TE
662
.A3
no.
FHWA-
RD-
76-
127

ort No. FHWA-RD-76-127

BAN TRAFFIC CONTROL SYSTEM TRAFFIC APTIVE NETWORK SIGNAL TIMING PROGRAM

Vol. 3. UTCS/TANSTP Data Base and Data Base Update Procedures; Offline Support Programs; Operating Procedures

Dept. of Transportation
MAR 14 1977

August 1976
Final Report

This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161

Prepared for
FEDERAL HIGHWAY ADMINISTRATION
Offices of Research & Development
Washington, D.C. 20590

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

The contents of this report reflect the views of De Leuw, Cather & Company which is responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policy of the Department of Transportation. This report does not constitute a standard, specification, or regulation.

The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein only because they are considered essential to the object of this document.

The report, in three volumes, is being distributed with five copies for each regional office, two copies for each division, and two copies for each State highway or transportation department. Direct distribution is being made to the division offices with sufficient copies to provide two for each State highway department so that the division offices may make the distribution to the cognizant State officials.

A limited number of additional copies of this report are available for official use and may be requested from Dr. W. W. Wolman, Chief, Traffic Systems Division, FHWA, HRS-33, Washington, D. C. 20590. These requests will be filled while the supply lasts. Additional copies for the public are available from the National Technical Information Service, Department of Commerce, 5285 Port Royal Road, Springfield, Virginia 22161. A small charge will be imposed for each copy ordered from NTIS.

TE
662
A3
no.
FHWA-
RD-
76-127

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.
FHWA-RD-76-127

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
Urban Traffic Control System Traffic Adaptive Network Signal Timing Program
Volume 3. UTCS/TANSTP Data Base and Data Base Update Procedures; Offline Support Programs; Operating Procedures

5. Report Date
August 1976

6. Performing Organization Code

7. Author(s)
R. W. Kessmann

8. Performing Organization Report No.

9. Performing Organization Name and Address
De Leuw, Cather & Company
Suite 604
1110 NASA Road One
Houston, Texas 77058

10. Work Unit No.
FCP 32B2-012

11. Contract or Grant No.
DOT-FH-11-7594

12. Sponsoring Agency Name and Address
Office of Research and Development
Federal Highway Administration
U. S. Department of Transportation
Washington, D. C. 20590

13. Type of Report and Period Covered
Final Report

14. Sponsoring Agency Code

Transportation
MAR 14 1977

15. Supplementary Notes
FHWA Contract Manager: Harry Lum (HRS-33)

16. Abstract
This document describes the Traffic Adaptive Network Signal Timing Program (TANSTP) of the Urban Traffic Control System. TANSTP is a second generation traffic control package wherein optimal traffic signal timing patterns are generated on-line as a function of current and predicted traffic flow conditions to minimize average network vehicle delay. In addition to the network optimization routine, the package includes a traffic flow predictor, real-time subnetwork determination routine, a subnetwork interfacing routine, a transition optimization routine, and a local intersection optimization routine which adjusts splits and offsets to minimize delay on a cycle by cycle basis. The TANSTP package uses the executive structure, the detector processing, and the controller command software of the first generation, UTCS software.

The report is contained in three volumes as follows:

Volume 1. Overview Description; Narrative Description and Flow Charts of TANSTP Routines

Volume 2. Data Base Interface and Support Subroutines; Modified First Generation Routines

Volume 3. UTCS/TANSTP Data Base and Data Base Update Procedures; Offline Support Programs; System Operating Procedures

17. Key Words
Signal System Optimization, Real Time Traffic Control Systems, Traffic Responsive Control Systems, Traffic Control Algorithms, Network Signal Systems, Second Generation Traffic Control Software

18. Distribution Statement
No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	203	

Form DOT F 1700.7 (8-69)

CONTENTS

VOLUME 1

		Page
1.	INTRODUCTION	1
2.	OVERVIEW DESCRIPTION OF THE UTCS/TANSTP SOFTWARE SYSTEM	4
2.1	Design Requirements and Features of UTCS/TANSTP SOFTWARE SYSTEM	4
2.2	Overview of the UTCS/TANSTP Software and Description of Interface Between the First and Second Generation Software	5
2.3	UTCS/TANSTP Routines and Subroutines; Functions and Computer Resource Requirements	10
2.4	Organization of UTCS CPU Functions	25
2.5	Priority Interrupt System	25
2.6	UTCS/TANSTP Software File Organization	28
2.7	UTCS/TANSTP Overlay Structure	28
3.	NARRATIVE DESCRIPTION AND FLOW CHARTS OF TANSTP ROUTINES	32
3.1	Routine MACRO	37
3.1.1	Purpose	37
3.1.2	Method	37
3.1.3	Cross Reference	38
3.2	Routine RTN5	51
3.2.1	Purpose	51
3.2.2	Method	51
3.2.3	Cross Reference	52
3.3	Routine PREDICT	59
3.3.1	Purpose	59
3.3.2	Method	59
3.3.3	Cross Reference	62
3.4	Routine RTSND	75
3.4.1	Purpose	75
3.4.2	Method	75
3.4.3	Cross Reference	81
3.5	Routine LSTSQS	141
3.5.1	Purpose	141
3.5.2	Method	141
3.5.3	Cross Reference	147

CONTENTS (Continued)

Page

3.6 Routine INTFC 185

3.6.1 Purpose 185
3.6.2 Method 185
3.6.3 Cross Reference 187

3.7 Routine TRNPAR 206

3.7.1 Purpose 206
3.7.2 Method 206
3.7.3 Cross Reference 207

3.8 Routine LOCAL 221

3.8.1 Purpose 221
3.8.2 Method 221
3.8.3 Cross Reference 225

VOLUME 2

4. DATA BASE INTERFACE AND SUPPORT SUBROUTINES . . . 1

4.1 SUBRT1 File Subroutines 6

4.1.1 Subroutine PERIO5 (Cycle Length Modulo) . . 7

4.1.1.1 Purpose 7
4.1.1.2 Calling Sequence 7
4.1.1.3 Method 7
4.1.1.4 Cross Reference 7

4.1.2 Subroutine LLGTHF (Load Link Length, Phase Code, and Weight) 9

4.1.2.1 Purpose 9
4.1.2.2 Calling Sequence 9
4.1.2.3 Method 9
4.1.2.4 Cross Reference 9

4.2 SUBRT2 File Subroutines 11

4.2.1 Subroutine GOMTRY (Link Geometry) 12

4.2.1.1 Purpose 12
4.2.1.2 Calling Sequence 12
4.2.1.3 Method 13
4.2.1.4 Cross Reference 13

4.2.2 Subroutine GSPLEL (Get Time-of-Day Split for LOCAL) 28

4.2.2.1 Purpose 28
4.2.2.2 Calling Sequence 28
4.2.2.3 Method 28
4.2.2.4 Cross Reference 28

CONTENTS (Continued)

		Page
4.2.3	Subroutine GSPLWL (Get Working Split for LOCAL)	30
	4.2.3.1 Purpose	30
	4.2.3.2 Calling Sequence	30
	4.2.3.3 Method	30
	4.2.3.4 Cross Reference	30
4.2.4	Subroutine LDMOE (Load Measures of Effectiveness Data)	32
	4.2.4.1 Purpose	32
	4.2.4.2 Calling Sequence	32
	4.2.4.3 Method	32
	4.2.4.4 Cross Reference	32
4.2.5	Subroutine PKWRK (Pack Working Offsets)	34
	4.2.5.1 Purpose	34
	4.2.5.2 Calling Sequence	34
	4.2.5.3 Method	34
	4.2.5.4 Cross Reference	34
4.2.6	Subroutine SPLPKL (Split Pack from LOCAL)	36
	4.2.6.1 Purpose	36
	4.2.6.2 Calling Sequence	36
	4.2.6.3 Method	36
	4.2.6.4 Cross Reference	36
4.2.7	Subroutine UNPAK (Unpack Controller Phasing Data)	38
	4.2.7.1 Purpose	38
	4.2.7.2 Calling Sequence	38
	4.2.7.3 Method	39
	4.2.7.4 Cross Reference	39
4.2.8	Subroutine UNPK1 (Unpack Link Data)	43
	4.2.8.1 Purpose	43
	4.2.8.2 Calling Sequence	43
	4.2.8.3 Method	44
	4.2.8.4 Cross Reference	44
4.2.9	Subroutine WORK (Working Offset Unpack)	48
	4.2.9.1 Purpose	48
	4.2.9.2 Calling Sequence	48
	4.2.9.3 Method	48
	4.2.9.4 Cross Reference	48

CONTENTS (Continued)

			Page
4.3	SUBRT3 File Subroutines		50
	4.3.1	Subroutine CORRECT (Idealized Offset Corrections)	51
		4.3.1.1 Purpose	51
		4.3.1.2 Calling Sequence	51
		4.3.1.3 Method	51
		4.3.1.4 Cross Reference	56
	4.3.2	Subroutine DELDAT (Obtain Delay Data)	64
		4.3.2.1 Purpose	64
		4.3.2.2 Calling Sequence	64
		4.3.2.3 Method	64
		4.3.2.4 Cross Reference	64
	4.3.3	Subroutine FMODE (Determine Mode)	68
		4.3.3.1 Purpose	68
		4.3.3.2 Calling Sequence	68
		4.3.3.3 Method	68
		4.3.3.4 Cross Reference	68
	4.3.4	Subroutine IOCHK (Check on I/O in Progress)	70
		4.3.4.1 Purpose	70
		4.3.4.2 Calling Sequence	70
		4.3.4.3 Method	70
		4.3.4.4 Cross Reference	70
	4.3.5	Subroutine LOADAT (Load Traffic Responsive Pattern Data)	72
		4.3.5.1 Purpose	72
		4.3.5.2 Calling Sequence	72
		4.3.5.3 Method	72
		4.3.5.4 Cross Reference	72
	4.3.6	Subroutine PATGENT (Pattern Data Generation)	97
		4.3.6.1 Purpose	97
		4.3.6.2 Calling Sequence	97
		4.3.6.3 Method	97
		4.3.6.4 Cross Reference	98
	4.3.7	Subroutine SHTDWN (Shutdown)	124
		4.3.7.1 Purpose	124
		4.3.7.2 Calling Sequence	124
		4.3.7.3 Method	124
		4.3.7.4 Cross Reference	124

CONTENTS (Continued)

Page

4.3.8 Subroutine SMINT (Sum Intervals) 126

4.3.8.1 Purpose 126
4.3.8.2 Calling Sequence 126
4.3.8.3 Method 126
4.3.8.4 Cross Reference 126

4.4 RTN5F File Subroutines 130

4.4.1 Subroutine LDMOE5 (Load Measures of Effectiveness for Level 5) 131

4.4.1.1 Purpose 131
4.4.1.2 Calling Sequence 131
4.4.1.3 Method 131
4.4.1.4 Cross Reference 131

4.4.2 Subroutine LINKD (Subnetwork Mapping) 134

4.4.2.1 Purpose 134
4.4.2.2 Calling Sequence 134
4.4.2.3 Method 134
4.4.2.4 Cross Reference 134

4.4.3 Subroutine RTNFT (TANSTP Surveillance Computations) 142

4.4.3.1 Purpose 142
4.4.3.2 Calling Sequence 142
4.4.3.3 Method 142
4.4.3.4 Cross Reference 142

4.5 PREDICT File Subroutines 150

4.5.1 Subroutine BUFFEROUT/BUFFERIN (Write or Retrieve RAD Data) 151

4.5.1.1 Purpose 151
4.5.1.2 Calling Sequence 151
4.5.1.3 Method 151
4.5.1.4 Cross Reference 152

4.5.2 Subroutine PRPRED (Print Predictor Statistics) 155

4.5.2.1 Purpose 155
4.5.2.2 Calling Sequence 155
4.5.2.3 Method 155
4.5.2.4 Cross Reference 156

CONTENTS (Continued)

Page

4.6 RTSND File Subroutines 163

4.6.1 Subroutine GETSPLT (Get Split) 164

4.6.1.1 Purpose 164
4.6.1.2 Calling Sequence 164
4.6.1.3 Method 164
4.6.1.4 Cross Reference 164

4.6.2 Subroutine GSACBC (Get Sum of Fixed Intervals in Overlapping Phases). 166

4.6.2.1 Purpose 166
4.6.2.2 Calling Sequence 166
4.6.2.3 Method 166
4.6.2.4 Cross Reference 166

4.6.3 Subroutine GTSMT (Get Sum of Fixed Intervals not Part of A-, B-, C-Phase Green) 168

4.6.3.1 Purpose 168
4.6.3.2 Calling Sequence 168
4.6.3.3 Method 168
4.6.3.4 Cross Reference 168

4.6.4 Subroutine IRTN (Map Core Groups) 170

4.6.4.1 Purpose 170
4.6.4.2 Calling Sequence 170
4.6.4.3 Method 170
4.6.4.4 Cross Reference 170

4.6.5 Subroutine LNKFL (Link Failure) 173

4.6.5.1 Purpose 173
4.6.5.2 Calling Sequence 173
4.6.5.3 Method 173
4.6.5.4 Cross Reference 173

4.6.6 Subroutine LPHASV (Load Link Phasing for Volumes). 175

4.6.6.1 Purpose 175
4.6.6.2 Calling Sequence 175
4.6.6.3 Method 176
4.6.6.4 Cross Reference 176

4.6.7 Subroutine SMSCYL (Return Sums of Fixed Intervals in Phase and Entry and Time-of-Day Cycle Length) 181

4.6.7.1 Purpose 181
4.6.7.2 Calling Sequence 181
4.6.7.3 Method 181
4.6.7.4 Cross Reference 181

CONTENTS (Continued)

Page

4.6.8 Subroutine SPLPK (Pack Splits) 183

4.6.8.1 Purpose 183
4.6.8.2 Calling Sequence 183
4.6.8.3 Method 183
4.6.8.4 Cross Reference 183

4.6.9 Subroutine SPLTWP (Pack Time-of-Day Split) 185

4.6.9.1 Purpose 185
4.6.9.2 Calling Sequence 185
4.6.9.3 Method 185
4.6.9.4 Cross Reference 185

4.6.10 Subroutine TOFAZE (Allocate Green Time to Two Phases) 187

4.6.10.1 Purpose 187
4.6.10.2 Calling Sequence 187
4.6.10.3 Method 187
4.6.10.4 Cross Reference 187

4.7 INTFC File Subroutines. 189

4.7.1 Subroutine ENTCYCL (Entry Cycle). 190

4.7.1.1 Purpose 190
4.7.1.2 Calling Sequence 190
4.7.1.3 Method 190
4.7.1.4 Cross Reference 190

4.7.2 Subroutine GETCYCLC (Get Cycle) 192

4.7.2.1 Purpose 192
4.7.2.2 Calling Sequence 192
4.7.2.3 Method 192
4.7.2.4 Cross Reference 192

4.7.3 Subroutine GETEO (Get Entry Offset). 194

4.7.3.1 Purpose 194
4.7.3.2 Calling Sequence 194
4.7.3.3 Method 194
4.7.3.4 Cross Reference 194

4.7.4 Subroutine GETEOFF (Get Entry Offset). . . . 196

4.7.4.1 Purpose 196
4.7.4.2 Calling Sequence 196
4.7.4.3 Method 196
4.7.4.4 Cross Reference 196

CONTENTS (Continued)

		Page
4.7.5	Subroutine GETINT (Get Intersections)	198
4.7.5.1	Purpose	198
4.7.5.2	Calling Sequence	198
4.7.5.3	Method	198
4.7.5.4	Cross Reference	198
4.7.6	Subroutine GETWORK (Get Working Offset)	200
4.7.6.1	Purpose	200
4.7.6.2	Calling Sequence	200
4.7.6.3	Method	200
4.7.6.4	Cross Reference	200
4.7.7	Subroutine LKCROS (Link Cross Reference)	202
4.7.7.1	Purpose	202
4.7.7.2	Calling Sequence	202
4.7.7.3	Method	202
4.7.7.4	Cross Reference	202
4.7.8	Subroutine MACA (Major Entry Array)	204
4.7.8.1	Purpose	204
4.7.8.2	Calling Sequence	204
4.7.8.3	Method	204
4.7.8.4	Cross Reference	204
4.7.9	Subroutine PAKF (Pack Transition Flags)	206
4.7.9.1	Purpose	206
4.7.9.2	Calling Sequence	206
4.7.9.3	Method	206
4.7.9.4	Cross Reference	206
4.7.10	Subroutine PUTEOFF (Put Entry Offset)	208
4.7.10.1	Purpose	208
4.7.10.2	Calling Sequence	208
4.7.10.3	Method	208
4.7.10.4	Cross Reference	208
4.8	LOCAL File Subroutines	210
4.8.1	Subroutine PERIOD (Modulo Routine)	211
4.8.1.1	Purpose	211
4.8.1.2	Calling Sequence	211
4.8.1.3	Method	211
4.8.1.4	Cross Reference	211

CONTENTS (Continued)

Page

4.8.2 Subroutine TOFAZ1 (Compute Split for Two Controller Phases) 213

4.8.2.1 Purpose 213
4.8.2.2 Calling Sequence 213
4.8.2.3 Method 213
4.8.2.4 Cross Reference 213

5. MODIFIED FIRST GENERATION ROUTINES 215

5.1 Routine A (RTNA - UTCS Executive) 217

5.1.1 Purpose 217
5.1.2 Discussion of Modifications 217

5.2 Routine C (RTNC - Once Per Vehicle) 230

5.2.1 Purpose 230
5.2.2 Discussion of Modifications 230

5.3 Routine H (RTNH - Fifteen-Minute Performance and Status Report) 233

5.3.1 Purpose 233
5.3.2 Discussion of Modifications 233

5.4 Routine I (RTNI - End-of-Day Report) 238

5.4.1 Purpose 238
5.4.2 Discussion of Modifications 238

5.5 Routine K (RTNK - CRT Formatting) 239

5.5.1 Purpose 239
5.5.2 Discussion of Modifications 239

5.6 Routine M (RTNM - Manual Control Panel) 243

5.6.1 Purpose 243
5.6.2 Discussion of Modifications 243

5.7 Routine Q (RTNQ - Shutdown) 245

5.7.1 Purpose 245
5.7.2 Discussion of Modifications 245

5.8 Routine S (RTNS - Controller Transition Computations) 248

5.8.1 Purpose 248
5.8.2 Discussion of Modifications 248

5.9 Routine T (RTNT - Interval Advance Routine) 249

5.9.1 Purpose 249
5.9.2 Discussion of Modifications 249

CONTENTS (Continued)

Page

5.10 Routine V (RTNV - Critical Intersection Computations) 250

5.10.1 Purpose 250
5.10.2 Discussion of Modifications 250

5.11 Routine W (RTNW - Controller Parameter Selection) . . 259

5.11.1 Purpose 259
5.11.2 Discussion of Modifications 259

5.12 Routine X (RTNX - Semiactuated Controller Computations) 271

5.12.1 Purpose 271
5.12.2 Discussion of Modifications 271

5.13 Routine Y (RTNY - Start-Up) 272

5.13.1 Purpose 272
5.13.2 Discussion of Modifications 272

VOLUME 3

6. DESCRIPTION OF UTCS/TANSTP DATA BASE AND DATA BASE UPDATE PROCEDURES 1

6.1 Definition of Data Base Parameters. 1

6.2 Basic Elements of UTCS/TANSTP Data Base and Graphical Representation of UTCS/TANSTP Data Base for Washington, D. C., UTCS Network 121

6.2.1 Controllers 121
6.2.2 Detectors 121
6.2.3 First Generation Links 121
6.2.4 Second Generation Links 122
6.2.5 Groups 122
6.2.6 Sections 122
6.2.7 Subnetwork Interface. 123

6.2.7.1 Group Interfaces 123
6.2.7.2 Section Interfaces. 123

6.2.8 Graphical Representation of UTCS/TANSTP Data Base for Washington, D. C., UTCS Network. 124

6.3 UTCS/TANSTP Data Base Update Procedures 131

6.3.1 Procedures for Adding a First Generation Link. 132
6.3.2 Procedures for Adding a Second Generation Link. 137
6.3.3 Procedures for Adding a Controller. 139
6.3.4 Procedures for Changing Group Assignment for a Controller 142

CONTENTS (Concluded)

Page

6.3.5 Procedures for Changing Section Assignment for a Controller. 143

7. OFF-LINE SUPPORT PROGRAMS. 144

7.1 PATGEN Program 144

7.1.1 PATGEN Program Data Arrays 145

7.2 Surveillance Data Dump Program 146

7.3 Predictor File Initialization Program 146

7.4 Predictor Parameter Generation Program (PARAM) . . 147

7.5 Binary Save Deck. 178

7.6 Software System Link and Load Deck. 178

8. UTCS/TANSTP SYSTEM OPERATING PROCEDURES AND SOFTWARE SYSTEM UPDATE PROCEDURES. 180

8.1 UTCS/TANSTP System Operating Procedures. 180

8.2 UTCS/TANSTP Software System Update Procedures. . . 182

8.2.1 Procedure to Update File Contained on Cards. 182

8.2.2 Procedure to Update File Contained on Compressed Source Tape (File UTCS) 182

8.2.3 Procedure to Update Pattern Data Via PATGEN. 183

8.2.4 Procedure to Initialize Predictor Data Files 183

REFERENCES. 184

ILLUSTRATIONS

		Page
	VOLUME 1	
1	Overview of TANSTP Optimization Software	6
2	Overview of Integrated UTCS/TANSTP Software System	8
3	UTCS/TANSTP Software; Overview Flow of Signal Time Keeping Data	11
4	UTCS/TANSTP Software System Overlay Structure	31
5	Routine MACRO	39
6	Routine RTN5	53
7	Routine PREDICT	63
8	Routine RTSND	82
9	Simple Network Demonstrating Continuity Links	144
10	Descriptors L, M, N, O for Link Connecting Node M to Node N	145
11	Routine LSTSQS	148
12	Routine INTFC	188
13	Routine TRNPAR	208
14	LOCAL Intersection Algorithm Link and Intersection Numbering Scheme	223
15	Routine LOCAL	227
	VOLUME 2	
16	Subroutine PERIO5	8
17	Subroutine LLGTHF	10
18	Subroutine GOMTRY	14
19	Subroutine GSPLEL	29
20	Subroutine GSPLWL	31
21	Subroutine LDMOE	33
22	Subroutine PKWRK	35

ILLUSTRATIONS (Continued)

		Page
23	Subroutine SPLPKL	37
24	Subroutine UNPAK	40
25	Subroutine UNPK1	45
26	Subroutine WORK	49
27	Subroutine CORRECT	57
28	Subroutine DELDAT	65
29	Subroutine FMODE	69
30	Subroutine IOCHK	71
31	Subroutine LOADAT	73
32	UTCS Controller Types; Pictorial Representation of Controller Parameters	99
33	Subroutine PATGENT	102
34	Subroutine SHTDWN	125
35	Subroutine SMINT	127
36	Subroutine LDMOE5	132
37	Subroutine LINKD	135
38	Tape Buffer Format	143
39	Subroutine RTNFT	144
40	Subroutine BUFFEROUT/BUFFERIN	153
41	Subroutine PRPRED	157
42	Sample Printout - Predictor Shutdown Report	162
43	Subroutine GETSPLT	165
44	Subroutine GSACBC	167
45	Subroutine GTSMT	169
46	Subroutine IRTN	171

ILLUSTRATIONS (Continued)

		Page
47	Subroutine LNKFL	174
48	Subroutine LPHASV	177
49	Subroutine SMSCYL	182
50	Subroutine SPLPK	184
51	Subroutine SPLTWP	186
52	Subroutine TOFAZE	188
53	Subroutine ENTCYCL	191
54	Subroutine GETCYCLC	193
55	Subroutine GETEO	195
56	Subroutine GETEOFF	197
57	Subroutine GETINT	199
58	Subroutine GETWORK	201
59	Subroutine LKCROS	203
60	Subroutine MACA	205
61	Subroutine PAKF	207
62	Subroutine PUTEOFF	209
63	Subroutine PERIOD	212
64	Subroutine TOFAZ1	214
65	Routine A	218
66	Routine C	231
67	Routine H	234
68	TANSTP Traffic Signal Timing Report	235
69	UTCS/TANSTP Intersection Status Display Format	240
70	UTCS/TANSTP System Status Display Format	242

ILLUSTRATIONS (Concluded)

Page

71 Routine Q 246

72 Routine V 251

73 Routine W 261

74 Routine Y 273

VOLUME 3

75 Controllers and Links in UTCS Network Group 1 125

76 Controllers and Links in UTCS Network Group 2 126

77 Controllers and Links in UTCS Network Group 3 127

78 Controllers and Links in UTCS Network Group 4 128

79 Controllers and Links in UTCS Network Group 5 129

80 Controllers and Links in UTCS Network Group 6 130

81 Parameter Generation Program - Hierarchy of Subroutines. . 148

82 Program PARAM 151

83 Subroutine FORIER. 156

84 Subroutine MLTREG 157

85 Subroutine BETA 159

86 Subroutine GAMMA. 160

87 Subroutine MEAN 161

88 Subroutine MEMSET 162

89 Subroutine MEMSTI 163

TABLES

VOLUME 1

Page

1. Functions of UTCS/TANSTP MACRO and MICRO Loops . . . 7

2. UTCS/TANSTP Routines and Subroutines; Functions, Priority Level, Programming Language, and Core Storage Requirements 12

3. Sigma 5 Interrupt Structure (External Group 2) 26

4. UTCS/TANSTP Interrupt Triggering Structure 27

5. UTCS/TANSTP File Names and Functions 29

6. Routines/Subroutines in Each File and Form of Source Input 30

7. MACRO Loop Optimization Decision Logic 34

8. Definition of Variable Names Used in Routine MACRO 44

9. Definition of Variable Names Used in RTN5 57

10. Definition of Variable Names Used in PREDICT. 71

11. Washington, D. C. UTCS Network; Instrumented Controllers in Each Group 78

12. Routine RTSND Split Change Decision Logic 80

13. Definition of Variable Names Used in Routine RTSND. 129

14. Definition of Variable Names Used in Routine LSTSQS 174

15. Definition of Variable Names Used in Routine INTFC 201

16. Definition of Variable Names Used in Routine TRNPAR. . . . 218

17. LOCAL Control Decision Logic. 222

18. Definition of Variable Names Used in Routine LOCAL. 263

VOLUME 2

19. Index to UTCS/TANSTP Subroutines 2

20. Definition of Link Phasing Codes and Link Delay Function as Function of Code 52

21. Linear Transformations as a Function of Controller Type for Obtaining B- and C-Phase Offsets as Function of A-Phase Offset 54

TABLES (Concluded)

		Page
22.	Definition of Variable Names Used in Subroutine PRPRED. .	160
	VOLUME 3	
23.	UTCS/TANSTP Data Base Index	3
24.	First Generation, UTCS, Static Arrays	22
25.	Second Generation, TANSTP, Static Arrays	38
26.	First Generation, UTCS, Dynamic Arrays	48
27.	Second Generation, TANSTP, Dynamic Arrays	102
28.	COMMON Blocks in UTCS/TANSTP Software	118
29.	Off-Line Support Programs and Form of Source	144
30.	Input Data Formats - Program PARAM	164
31.	Definition of Variable Names Used in Program PARAM . . .	165
32.	Definition of Variable Names Used in Subroutine FORIER . .	169
33.	Definition of Variable Names Used in Subroutine MLTREG.	171
34.	Definition of Variable Names Used in Subroutines BETA and GAMMA	173
35.	Definition of Variable Names Used in Subroutine MEAN . . .	175
36.	Definition of Variable Names Used in Subroutine MEMSET.	176
37.	Definition of Variable Names Used in Subroutine MEMST1. .	177

6. DESCRIPTION OF UTCS/TANSTP DATA BASE AND DATA BASE UPDATE PROCEDURES

This section describes the data base used to support the UTCS/TANSTP software system. Subsection 6.1 provides a definition of each mnemonic used in the data base. Subsection 6.2 presents a graphical representation of the data base numbering scheme used for controllers and links in the application of the software to the Washington, D. C. UTCS traffic signal network. Subsection 6.3 describes data base update procedures required to be performed by the traffic engineer/programmer analyst to accommodate changes in the composition of the traffic signal network to be controlled by the UTCS/TANSTP software system.

6.1 DEFINITION OF DATA BASE PARAMETERS

The data base contains two types of data. These are:

1) Static data.

2) Dynamic data.

The static data arrays are assembled or compiled into the software system prior to run time and remain fixed during the execution of the software system. These arrays contain information specifying the traffic signal network characteristics. Examples of static data are controller type, link to controller assignment, controller phase association of each link, link length, controller to section assignment, detector to link assignment, and link to section assignment. Also included in this category are weighting and conversion constants. The static arrays must be updated by the traffic engineer/programmer analyst and the software must be reassembled if detectors, controllers, links, or sections are to be added or deleted from the network controlled by the UTCS/TANSTP software.

The dynamic data are generally initialized to zero at system startup and are changed continually by the software programs during their execution. Examples of this type of data are traffic flow data (volumes, speeds, occupancies) and event timers. In changing the traffic signal characteristics, the programmer/analyst need only to worry that enough core space be reserved for these arrays to accommodate the network to be controlled by the software. The reserve limits are either assembled or compiled into the software system prior to execution time.

These types of data can also be subdivided in the UTCS/TANSTP software as follows:

1) UTCS First Generation arrays

2) TANSTP Second Generation arrays

The first generation arrays are those which existed in the UTCS software prior to the integration of the second generation, TANSTP software. The second generation arrays were added to the UTCS software to support the on-line traffic responsive pattern generation capability.

Table 23 of this section lists each data base parameter in the UTCS/TANSTP software and gives its classification in terms of whether it is static, dynamic, first, or second generation. This table also provides indexing to one of four tables in this section which gives the definition of each parametric array, the index in the software which is used to access the array, the routine or subroutine which defines or allocates the array, the units, the scale factor (as scale of BX indicates the binary point is X bits from the least significant end of the computer word), and the format of the data in the array.

Tables 24 and 25 apply to first and second generation static arrays, respectively. Tables 26 and 27 apply the first and second generation dynamic arrays, respectively.

Finally, Table 28 lists all blocked COMMON blocks compiled into the UTCS/TANSTP software. (No blank COMMON is used.) Given are the dimensions of each common block, the primary function of each block, the parameters included in each block, and the routines which use each block.

Table 23. UTCS/TANSTP Data Base Index Table

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
ABBCABCG	Dynamic	Second	27
ABCEND	Static	Second	25
ABCSTRT	Static	Second	25
ADVINH	Dynamic	First	26
AEXT	Static	Second	25
AGR	Dynamic	First	26
AHCC	Dynamic	First	26
AHHIST	Dynamic	First	26
AMOEPN	Static	First	24
AOG	Dynamic	First	26
APO	Dynamic	First	26
AVSP	Dynamic	First	26
AVSPP	Dynamic	Second	27
BIAS	Dynamic	Second	27
BYTCNT	Dynamic	First	26
CC1	Dynamic	First	26
CC2	Dynamic	First	26
CC2P	Dynamic	Second	27
CGRPN	Static	Second	25
CHGTABLE	Dynamic	First	26
CHNGTH	Static	Second	25
CICBYP	Dynamic	First	26
CICCNT	Dynamic	First	26
CICEF	Dynamic	First	26
CICINH	Dynamic	Second	27
CICSCT	Static	First	24
CICSF	Dynamic	First	26
CICSYS	Static	First	24
CICTF	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
CICTODF	Dynamic	First	26
CIRCR	Dynamic	First	26
CMALF	Dynamic	First	26
CMITFT	Static	Second	25
CMMALF	Dynamic	First	26
CNTDR	Dynamic	First	26
COMKQ1	Dynamic	First	26
COMKQ2	Dynamic	First	26
COMKV0	Dynamic	First	26
COMLENG	Dynamic	First	26
COMTAB	Dynamic	First	26
CONDIS	Dynamic	First	26
CONLENG	Dynamic	First	26
CONSEC	Static	First	24
CONTAB	Dynamic	First	26
CONTLN	Static	First	24
CONTNM	Dynamic	First	26
CONTZN	Static	First	24
COUNT	Dynamic	Second	27
CPAOG	Dynamic	First	26
CPANIN	Dynamic	First	26
CPANOU	Dynamic	First	26
CPANOUT	Dynamic	First	26
CPFLG	Dynamic	First	26
CRTAC1	Dynamic	First	26
CRTAC2	Dynamic	First	26
CRTBF1	Dynamic	First	26
CRTBF2	Dynamic	First	26
CRTDC1	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
CRTDC2	Dynamic	First	26
CRTERR2	Dynamic	First	26
CRTFL1	Dynamic	First	26
CRTFL2	Dynamic	First	26
CRTNUM	Dynamic	First	26
CTLSEC	Static	First	24
CTLSYS	Static	First	24
CUROFF	Dynamic	First	26
CYC(I, J)	Dynamic	Second	27
CYCLC	Dynamic	First	26
CYCMAX	Static	Second	25
CYCMIN	Static	Second	25
D	Dynamic	First	26
DACC	Dynamic	First	26
DATE	Dynamic	First	26
DAY	Dynamic	First	26
DAYOWK	Dynamic	First	26
DBP	Dynamic	First	26
DCF	Dynamic	First	26
DELA	Dynamic	First	26
DELDAY	Dynamic	First	26
DELP	Dynamic	First	26
DELT	Dynamic	Second	27
DELWN(I)	Dynamic	Second	27
DELWO(I)	Dynamic	Second	27
DETLENG	Dynamic	First	26
DETM	Dynamic	Second	27
DETTAB	Dynamic	First	26
DFAIL	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
DIS	Dynamic	First	26
DM	Dynamic	First	26
DMALF	Dynamic	First	26
DMODE	Dynamic	First	26
DOF	Dynamic	First	26
DOFF	Dynamic	First	26
DOW	Dynamic	First	26
DPATNUM	Dynamic	First	26
DSGSF	Dynamic	First	26
DSPA	Static	First	24
DT	Static	First	24
DWL	Static	First	24
DWLP	Static	First	24
EBCMODE	Dynamic	First	26
EMAST	Dynamic	First	26
ENTCYCLC	Dynamic	Second	27
ENTCYSV	Dynamic	Second	27
ENTFLG	Dynamic	First	26
ENTRYINT	Dynamic	First	26
ENTRYOFF	Dynamic	First	26
EODC	Dynamic	First	26
EPSILON	Dynamic	First	26
EPSOF	Dynamic	First	26
F	Dynamic	First	26
FAIL	Dynamic	First	26
FAILTAB	Dynamic	First	26
FAILTB	Dynamic	First	26
FFS	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
FLERFL	Dynamic	First	26
FMC	Dynamic	First	26
FMMARK	Dynamic	First	26
FMSECF	Dynamic	First	26
FMSYSF	Dynamic	First	26
FMTCHG	Dynamic	First	26
FMTNUM	Dynamic	First	26
FMTODC	Dynamic	First	26
FMTTYP	Dynamic	First	26
FPPF	Dynamic	First	26
FSETCF	Dynamic	First	26
FSETF	Dynamic	First	26
FSTFLG	Dynamic	First	26
FTAB	Dynamic	First	26
FTADWRD	Dynamic	First	26
GA	Dynamic	First	26
GA1	Dynamic	First	26
GA2	Dynamic	First	26
GAB	Dynamic	First	26
GABBCABC	Dynamic	First	26
GABCP	Dynamic	First	26
GABP	Dynamic	First	26
GAI	Dynamic	First	26
GAMIN	Dynamic	First	26
GAP	Dynamic	First	26
GAP1	Dynamic	First	26
GATEMP	Dynamic	First	26
GAW	Dynamic	First	26
GAWTEMP	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
GB	Dynamic	First	26
GBMIN	Dynamic	First	26
GBTEMP	Dynamic	First	26
GBW	Dynamic	First	26
GBWTEMP	Dynamic	First	26
GC	Dynamic	First	26
GCONT	Static	Second	25
GCONTSI	Static	Second	25
GMINABC4	Dynamic	First	26
GRIPFC	Static	Second	25
GRPNO	Static	Second	25
GSBNET	Dynamic	Second	27
GSECT	Static	Second	25
GT	Dynamic	First	26
HEADWY	Static	Second	25
HBUFLNG	Dynamic	First	26
HMAST	Dynamic	First	26
HOLD	Dynamic	First	26
HOLDTEMP	Dynamic	First	26
HOLIDF	Dynamic	First	26
HOLIDTBL	Dynamic	First	26
HSARR	Dynamic	First	26
I	Dynamic	First	26
IAA	Dynamic	Second	27
IADVP	Dynamic	First	26
IALPH	Static	Second	25
IARG	Static	Second	25
IB	Dynamic	First	26
IBL	Dynamic	First	26
IBRANIND	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
IC	Dynamic	First	26
ICGSI	Static	Second	25
IEA	Dynamic	Second	27
IL	Dynamic	First	26
ILL	Dynamic	Second	27
IM	Dynamic	First	26
IMAST	Dynamic	First	26
INCG	Static	Second	25
INFCR1	Static	Second	25
INHIBITE	Dynamic	First	26
INHT	Dynamic	First	26
INTERV2	Static	First	24
INTERVH	Static	First	24
INTF	Dynamic	Second	27
INTFO	Dynamic	Second	27
INTN	Static	Second	25
INTNS	Static	Second	25
INTN2	Static	Second	25
INTN2S	Static	Second	25
INT1SF	Dynamic	First	26
INT1SHF1	Dynamic	First	26
INT1SHF2	Dynamic	First	26
INTNUM	Dynamic	First	26
INTNUML	Dynamic	First	26
INTOUT	Dynamic	First	26
INTRNUM	Dynamic	First	26
IO	Dynamic	Second	27
IPRD	Static	Second	25

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
IRADRD	Dynamic	Second	27
IREOPT	Dynamic	Second	27
ISB(I, J)	Dynamic	Second	27
ISO	Dynamic	Second	27
ITRDLY	Dynamic	Second	27
I1	Dynamic	First	26
I2	Dynamic	First	26
JCOUNT	Dynamic	First	26
JJ	Dynamic	First	26
JMAST	Dynamic	First	26
K	Dynamic	First	26
KCL	Static	First	24
KCNT	Dynamic	First	26
KCSF	Dynamic	First	26
KCTC	Dynamic	First	26
KCTF	Dynamic	First	26
KHPS	Static	First	24
KMAST	Dynamic	First	26
KMCSP	Dynamic	First	26
KNEW	Static	First	24
KPCO	Static	First	24
KPW	Static	First	24
KQ1	Static	First	24
KQ2	Static	First	24
KS	Static	First	24
KSP	Static	First	24
KVPH	Static	First	24
KV0	Static	First	24
KV1	Static	First	24
KV2	Static	First	24
KV11	Static	Second	25

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
KW	Static	First	24
K3PH	Static	First	24
LAG	Dynamic	First	26
LCCRT	Static	First	24
LDSET	Static	First	24
LEAD	Dynamic	First	26
LENGTH	Static	Second	25
LEVL5O	Dynamic	Second	27
LEVL5A	Dynamic	Second	27
LEV5SF	Dynamic	Second	27
LFDPN	Static	First	24
LFZE	Static	Second	25
LINKDI	Static	First	24
LINKN	Static	Second	25
LINKNS	Static	Second	25
LINKPH	Static	First	24
LINKSI	Static	First	24
LL	Static	First	24
LL1	Static	Second	25
LMAP	Static	First	24
LMAST	Static	First	24
LMO	Static	Second	25
LNKARW	Static	First	24
LNKCON	Static	First	24
LNKPT	Static	First	24
LNKSPEED	Dynamic	First	26
LOCDAT	Static	Second	25
LOCFLG	Static	Second	25
LPBUF	Dynamic	First	26
LSI	Static	First	24
LSQ	Dynamic	Second	27

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
LPBUF	Dynamic	First	26
LSI	Static	First	24
LSQ	Dynamic	Second	27
MARR	Dynamic	First	26
MARR1	Dynamic	First	26
MASK	Static	First	24
MAST	Dynamic	First	26
MAXNC	Static	First	24
MAXNLKS	Static	First	24
MCDF	Dynamic	First	26
MI	Dynamic	First	26
MMAST	Dynamic	First	26
MNUMBF	Static	First	24
MODE$	Dynamic	First	26
MOEPSI	Static	First	24
MOE1P	Static	First	24
MOE1PRF	Static	First	24
MOE1RLN	Static	First	24
MOE2GLN	Static	First	24
MOE2P	Static	First	24
MOE2RLN	Static	First	24
MOETMP	Dynamic	First	26
MOETP2	Dynamic	First	26
MOE1CLK	Dynamic	First	26
MOE2CLK	Dynamic	First	26
MONTH	Dynamic	First	26
MOVTAB	Dynamic	First	26
MTHLD	Dynamic	First	26
MVPLUSKO	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
MOVTAB	Dynamic	First	26
MPOKEY	Dynamic	First	26
M1	Static	First	24
NAMOEP	Static	First	24
NC	Static	First	24
NCIGRP	Static	Second	25
NCLM1	Static	First	24
NCONTI	Static	First	24
NCT	Static	First	24
NCWRD	Static	First	24
NCWRD2	Static	First	24
NCWRD64	Static	First	24
NCYCLE	Dynamic	Second	27
NDALNK	Static	First	24
NDD	Dynamic	Second	27
NDOL	Static	First	24
NEXTPAGE	Dynamic	First	26
NGRIPF	Static	Second	25
NGRPI	Static	Second	25
NGRPIS	Static	Second	25
NGRPS	Static	Second	25
NHOLIDAY	Dynamic	First	26
NI	Dynamic	First	26
NICIGP	Static	Second	25
NII	Dynamic	Second	27
NINTF	Static	Second	25
NINTFS	Static	Second	25
NMAX	Static	Second	25
NOCONT	Static	First	24

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
NOFAIL	Dynamic	First	26
NONCON$	Static	First	24
NOT1NTOL	Dynamic	First	26
NPATBYT	Static	First	24
NPATBYT4	Static	First	24
NQL	Static	First	24
NREPORT	Dynamic	First	26
NRESPAT	Static	First	24
NSACC	Dynamic	First	26
NSDAY	Dynamic	First	26
NSECTM1	Static	First	24
NSTOPS	Dynamic	First	26
NSUTC	Dynamic	First	26
NSUBO	Dynamic	Second	27
NUMAX	Static	First	24
NUMCTL	Static	First	24
NUMLNK	Static	First	24
NUMMOV	Dynamic	First	26
NUMSCT	Static	First	24
NUMVAR	Static	First	24
NUMVED	Static	First	24
N2	Static	First	24
N2GCL	Static	Second	25
OACC	Dynamic	First	26
OCCPP	Dynamic	Second	27
OCCW	Static	Second	25
OCTH	Static	Second	25
ODAY	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
OPC	Dynamic	First	26
OPCP	Dynamic	Second	27
OPTIN	Dynamic	Second	27
OPTIME	Static	Second	25
P	Dynamic	First	26
PAGEKEY	Dynamic	First	26
PAGNUM	Dynamic	First	26
PAR	Static	First	24
PARCOD	Dynamic	First	26
PATNUM	Dynamic	First	26
PCOT	Dynamic	First	26
PDOF	Dynamic	First	26
PLS	Dynamic	First	26
PREVDATE	Dynamic	First	26
PRTFLG	Dynamic	First	26
PS	Dynamic	Second	27
PTN	Dynamic	First	26
PV	Dynamic	Second	27
PW	Dynamic	First	26
PWL	Dynamic	First	26
PW0	Dynamic	First	26
PW2	Dynamic	First	26
PW3	Dynamic	First	26
P1	Dynamic	First	26
P2	Dynamic	First	26
P3	Dynamic	First	26
P4	Dynamic	First	26
Q	Dynamic	First	26
QACC	Dynamic	First	26
QBAR	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
QCIC	Dynamic	First	26
QCICT	Dynamic	First	26
QCNT	Dynamic	First	26
ODAY	Dynamic	First	26
QG	Dynamic	First	26
QM	Dynamic	First	26
QMAST	Dynamic	First	26
QPT	Dynamic	First	26
QR	Dynamic	First	26
QRD	Dynamic	First	26
QVOK	Dynamic	First	26
Q1	Dynamic	First	26
RAWLNKC	Dynamic	First	26
RAWLNKO	Dynamic	First	26
RAWOCC	Dynamic	First	26
RAWOCP	Dynamic	Second	27
RAWSP	Dynamic	First	26
RAWSPP	Dyanmic	Second	27
RLNKC	Dynamic	Second	27
RLNKO	Dynamic	Second	27
RLSPEED	Dynamic	Second	27
RRABC1	Dynamic	First	26
R1C	Dynamic	First	26
R0, R1... ....,R15	Dynamic	First	26
S	Dynamic	First	26
SABC	Dynamic	Second	27
SACC	Dynamic	First	26
SACCP	Dynamic	Second	27

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
SATHLW	Static	Second	25
SATHRH	Static	Second	25
SATTHL	Static	Second	25
SATTHL	Static	Second	25
SAVRO	Dynamic	Second	27
SBAR	Dynamic	First	26
SDAY	Dynamic	First	26
SECCFC	Dynamic	First	26
SECCONSI	Static	First	24
SECCONT	Static	First	24
SECDFF	Static	Second	25
SECIFC	Static	Second	25
SECDFC	Dynamic	First	26
SECTLKN	Static	First	24
SEMMSK	Static	First	24
SEND	Dynamic	Second	27
SEND1	Dynamic	Second	27
SEND2	Dynamic	Second	27
SFADD	Static	Second	25
SHTFLG	Dynamic	First	26
SLFUPN	Dynamic	First	26
SMTABC	Dynamic	Second	27
SPEED	Dynamic	Second	27
SPLCHG	Dynamic	First	26
SPLCHT	Static	Second	25
SPLIT	Dynamic	Second	27
SPLITW	Dynamic	Second	27
SSFW	Dynamic	First	26
ST	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
STARTA	Dynamic	First	26
STARTUPF	Dynamic	First	26
STATE	Dynamic	First	26
STBTIM	Dynamic	First	26
STBYD	Dynamic	First	26
STFLG2	Dynamic	First	26
STFLG3	Dynamic	First	26
STFLG4	Dynamic	First	26
STFLG5	Dynamic	First	26
STOP1	Dynamic	First	26
STOP2	Dynamic	First	26
STPTEMP	Dynamic	First	26
SUMA	Dynamic	First	26
SUMABCT	Dynamic	First	26
SUMACBC	Dynamic	Second	27
SURDIS	Dynamic	First	26
SYSFLG	Dynamic	First	26
TAMODE	Dynamic	First	26
TBSIZE	Static	Second	25
TBUFF	Dynamic	Second	27
TCODE	Dynamic	First	26
TD	Dynamic	First	26
TDMODE	Dynamic	First	26
TEMP	Dynamic	First	26
TEMPSCF	Dynamic	First	26
TEMPMOEF	Dynamic	First	26
TEMPQ	Dynamic	First	26
TEMPS	Dynamic	First	26
TEMPQ	Dynamic	First	26
TEMPS	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
TEMPZ	Dynamic	First	26
TFAIL	Dynamic	First	26
THRESHOLD	Dynamic	First	26
THRESH	Dynamic	First	26
TIME	Dynamic	First	26
TIMEOUT	Dynamic	First	26
TIMMCH	Dynamic	First	26
TINDEX	Dynamic	Second	27
TK	Dynamic	First	26
TKM	Dynamic	First	26
TLENGTH	Dynamic	First	26
TMDY	Dynamic	First	26
TODC	Dynamic	First	26
TOL	Static	Second	25
TOSU	Dynamic	First	26
TP	Dynamic	First	26
TPAT	Dynamic	First	26
TRANSF	Dynamic	First	26
TRTM	Dynamic	First	26
TSECT	Dynamic	First	26
TST	Dynamic	First	26
TSTART	Dynamic	First	26
TT	Dynamic	First	26
TTC	Dynamic	First	26
TTIME	Dynamic	First	26
TTSA	Dynamic	First	26
TTSASI	Dynamic	First	26
TVPLUSKO	Dynamic	First	26
VACC	Dynamic	First	26

Table 23. UTCS/TANSTP Data Base Index Table (Continued)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
VACCP	Dynamic	Second	27
VARINT	Static	First	24
VDAT	Dynamic	First	26
VHIST	Dynamic	First	26
VO	Dynamic	First	26
VOL	Dynamic	First	26
VOT	Dynamic	First	26
VPHT	Dynamic	First	26
VPLUSKO	Dynamic	First	26
W	Dynamic	First	26
WD9	Static	First	24
WD900	Static	First	24
WHSF	Dynamic	First	26
WMAST	Dynamic	First	26
WMINF	Dynamic	First	26
WRKOFF	Dynamic	First	26
WSF	Dynamic	First	26
YIELD	Dynamic	First	26
ZD0	Dynamic	First	26
ZD1	Dynamic	First	26
ZF2	Dynamic	First	26
ZF3	Dynamic	First	26
ZLIM	Dynamic	First	26
ZOLD1	Dynamic	First	26
ZOLD2	Dynamic	First	26
ZSI	Static	First	24
ZSLIM	Static	First	24
Z2	Static	First	24

Table 23. UTCS/TANSTP Data Base Index Table (Concluded)

Mnemonic	Dynamic or Static Array	First or Second Generation Array	Applicable Definition Table
1GLNK	Static	Second	25
2GCLSI	Static	Second	25
IMINCT	Dynamic	First	26
2NCWRD	Static	First	24
5MINCT	Dynamic	Second	27
5MINF	Dynamic	Second	27
10VWD9	Dynamic	Second	27
15MINCYC	Dynamic	First	26
15TBOFF	Dynamic	First	26
15TBYTE	Static	First	24
15TBSIZE	Static	First	24
15TINDEX	Static	First	24

Table 24. First Generation, UTCS, Static Arrays

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
AMOEPN	Adjustable MOE period no. INDEX = 6* CONTR + (adjustable MOE periods counter)	INDEX	*RTNY, RTNG, RTNU	ND	B0	1 byte/MOE period
CICSCT	Possible CIC section wide flags. INDEX = (cont. no.) + 128* (section no.)	INDEX	*RTNY, RTNK, RTNM	ND	B0	1 bit/controller/ section
CICSYS	System wide CIC flags	CONTR	*RTNY, RTNK, RTNM	ND	B0	1 bit/controller**
CONSEC	Converts from controller number to actual section number	CONTR	*RTNY, RTNK, RTNM KFMT	ND	B0	1 bit/controller**
CONTLN	Link numbers for each controller pointed to by LSI multi-detector (critical links) are listed first for each controller	LSI	*RTNY, RTND, RTNK, RTNV, RTNW, LPHASV, UNPK1	ND	B0	1 halfword/link

*Defining routine.

**Word interleaved.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CONTZN	Zone numbers for each controller pointed to by ZSI	ZSI	*RTNY, RTNH, RTNI, RTNK, RTNM	ND	B0	1 byte/zone
CTLSEC	Section wide controller array	NA	*RTNY, RTNM	ND	B0	1 bit/controller/ section
CTLSYS	System wide controller array	NA	*RTNY, RTNM	ND	B0	1 bit/controller
DELA	Absolute value threshold for determining link out of tolerance	NA	RTNH	VPH	B0	1 word
DELP	Percent threshold for determining link out of tolerance	NA	RTNH	ND	B16	1 word
DSPA	Disk starting point address. INDEX = FMTODC	INDEX	*RTNY, RTNW	ND	B0	1 byte/15 minutes
DT	Array of delta time between predicted green time and the stops threshold	HALF-VEH	*RTNY, RTNC, RTND	SEC	B2	1 byte/DT

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
DWL	Converts from detector number to link number	DET	*RTNY, RTNC, RTNE, RTNK, KFMT, RTN5	ND	B0	1 halfword/ detector
DWLP	Detector position. Code is zero for only detector on link, 15 for dummy detectors	DET	*RTNY, RTNC, RTNE, RTN5	ND	B0	1 byte/detector
INTERV2	Second interval beyond end of A-Phase green	CONTR	*RTNY, RTNP, RTNT	ND	B0	1 byte/controller
INTERVH	Highest interval number	CONTR	*RTNY, RTNP, RTNT, RTNK, PATGENT, SMINT	ND	B0	1 byte/controller

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
KCL	Duration of each cycle length	CYCLC	*RTNY, RTNH, RTNK, RTNM, RTNN, RTNS, RTNU, RTNX, UNPAK, GOMTRY, DELDAT, LOADAT, PATGENT, SMSCYL, UNPK1, GETCYCLC, ENTCYCL, PUTEOFF, MACA	SEC	B1	1 word/cycle length
KHPS	1/3600 constant converts seconds to hours	NA	*RTNY, RTND	HRS/ SEC	B28	1 word
KNEW	Detector being processed	NA	*RTNY, RTNB, RTNC	ND	B0	1 word
KPCO	Percent occupancy smoothing filter gain	NA	RTNF	ND	B1	1 word/gain

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
KPW	Pulse duration smoothing constant	NA	RTNC	ND	B8	1 word
KQ1	Smoothing filter gain for QBAR at start of link green	NA	*RTNY, RTND	ND	B31	1 word
KQ2	Smoothing filter gain for QCIC at start of link green	NA	*RTNY, RTND	ND	B31	1 word
KS	Smoothing filter gain for stops	NA	*RTNY, RTND	ND	B31	1 word
KSP	Speed smoothing filter gain	NA	RTNF	ND	B1	1 word/gain
KVPH	Volume smoothing filter gain	NA	RTNF	ND	B1	1 word/gain
KV0	Smoothing filter gain for VO	NA	*RTNY, RTND	ND	B31	1 word
KV1	Converts reciprocal of pulse length (/SEC) to speed (MPH)	NA	*RTNF, RTNH	SEC^{-1}	B2	1 word
KV2	Conversion factor feet per second to miles per hour = 0.6818181	NA	RTNH	ND	B16	1 word
KW	Occupancy weighting factor	NA	*RTNY, RTNH	VEH/ HR	B0	1 word

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
K3PH	Type of CIC control. Code is: 0=2 critical movements and no overlapping (Type A); 1=3 Non-overlapping movements (Type B); 2=3 movements where the A and C Phases overlap (Type C); 3=3 movements with B and C Phases overlapping (Type D).	CONTR	*RTNY, RTNS, RTNV, LOADAT, PATGENT, GOMTRY, UNPAK, UNPK1, LPHASV, PUTEOFF, MACA	ND	B0	1 byte/controller
LCCRT	Equated to LNKCON					
LDSET	Link start entry	CONTR	*RTNY, RTNV, LPHASV, UNPK1	ND	B0	1 byte/controller
LFDPN	Length for desired pattern number. Number of words in pattern.	PAT NO.	*RTNY, RTNW	ND	B0	1 halfword/ section
LINKDI	Direction for each link: 0=N; 1=S; 2=E; 3=W; 4=NE; 5=NW; 6=SE; 7=SW; 8=EO	LINK	*RTNY, RTNH, RTNK	ND	B0	1 byte/link
LINKPH	Equated to LNKPT					

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine / Subroutine	Units	Scale	Format
LINKSI	First link index number for section in SECTLKN	SECT	*RTNY, RTNI, RTNW	ND	B0	1 halfword/ section
LL	Occupancy correction factor	NA	*RTNF, RTNH	ND	B16	1 word
LMAP	Label equated to MARR	NA	RTNJ	ND	NA	1 bit/status
LMAST	Priority of RTNL in MAST	NA	*RTNY, RTNA, RTNL	ND	B0	1 word
LNKARW	Link arrow display array	LINK	RTNJ	ND	NA	1 byte/link
LNKCON	Converts from link numbers to controller numbers	LINK	*RTNY, RTND, RTNH, RTNI, RTNK, RTNM, KFMT	ND	B0	1 byte/link
LNKPT	Link phase table. Phase in which traffic flows. Codes are: A1=000; A2=001; B1=010; B2=011; C=100; D=110	LINK	*RTNY, RTND, RTNK	ND	B0	1 byte/link

*Defining routine.

Table 24. First Generation, UTCS, Status Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
LSI	First link index number for each controller in array CONTLN	CONTR	*RTNY, RTNK, RTNV, RTNW, LPHASV, UNPK1	ND	B0	1 halfword/ controller
MASK	Thirty-two one bit masks. MASK(K) has the kth bit equal '1' and all other bits equal '0'	BIT INDEX	*RTNN, RTNA, RTND, RTNH, RTNK, RTNL, RTNM, RTNI, RTNT, RTNW, RTNX, RTNY, KFMT, IOCHK, LOADAT, SPLPK, PAKF, RTN5	ND	NA	32 words (1 word/ mask)
MAXNC	Equated to NUMCTL					
MAXNLKS	Equated to NUMLNK					

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
MNUMBF	Maximum number of failures in FAILTAB	NA	*RTNY, RTNH, RTNK, KFMT	ND	B0	1 word
MOEPSI	Location of first MOE period	CONTR	*RTNY, RTND, RTNG, RTNU	ND	B0	1 halfword/ controller
MOE1P	MOE1 period durations for each controller	MOEP-SI	*RTNY, RTNG, RTNU	SEC	B1	1 byte/MOE1 period
MOE1RLN	MOE1 red link number. Number of link which will have downstream red at end of present MOE1 period	MOE1 PER	*RTNY, RTND	ND	B0	1 halfword/MOE period
MOE2GLN	MOE2 green link number link number having green or amber signal at end of present MOE2 period	MOE-PER	*RTNY, RTND	ND	B0	1 halfword/MOE period
MOE2P	MOE2 period durations for each controller	MOEP-SI	*RTNY, RTND, RTNG, RTNU	SEC	B1	1 byte/MOEPSI
MOE2RLN	MOE2 red link number. Link number having red signal at end of present MOE2 period	MOE-PER	*RTNY, RTND	ND	B0	1 halfword/MOE period

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
M1	Multiplying factor of normal detector gap length used for isolating detector obstruction or malfunction	NA	*RTNY, RTNE	VEH	B6	1 word
NAMOEP	Number of adjustable MOE periods	CONTR	*RTNY, RTNG, RTNU	ND	B0	1 byte/controller
NC	Number of controllers minus unity	SECT	*RTNY, RTNW	ND	B0	1 byte/section
NCLM1	Number of cycle lengths minus 1	NA	RTNN	ND	B0	1 word
NCONTI	Number of controller intervals in corresponding adjustable MOE period. INDEX = adjustable MOE period number + 6* (controller number)	INDEX	*RTNY, RTNG	ND	B0	6 bytes/controller
NCT	Maximum possible raw car count of a normal detector	NA	RTNE	VEH	B0	1 word
NCWRD	Number of words in a controller bit array	NA	*RTNY, RTNM	ND	B0	1 word
NCWRD2	Twice the number of words in a controller bit array	NA	*RTNY, RTNM	ND	B0	1 word

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
NCWRD64	Sixty-four times the number of words in a controller bit array	NA	*RTNY, RTNM	ND	B0	1 word
NDALNK	Number of detectors assigned to link including dummies	LINK	*RTNY, RTNE, RTN5	ND	B0	1 byte/link
NUMAX	Equated to NUMVED					
NDOL	Number of detectors instrumented on link	LINK	*RTNY, RTND, RTNF, RTNH, RTNI, RTNK, RTN5	ND	B0	1 byte/link
NOCONT	List of nonexisting controllers	NA	*RTNK, RTNH	ND	B0	1 byte/ nonexistent controller
NONCON$	Number of nonexisting controllers	NA	*RTNK, RTNH	ND	B0	1 word
NPATBYT	NRESPAT + NMAMPAT	NA	*RTNY, RTNM, RTNW	ND	B0	1 word

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
NPATBYT4	4 * NPATBYT	NA	*RTNY, RTNM, RTNW	ND	B0	1 word
NQL	Number of multidetector links	CONTR	*RTNY, RTNK, RTNV, LPHASV	ND	B0	1 byte/controller
NRESPAT	Number of traffic responsive pattern numbers in each history buffer	NA	*RTNY, RTNM, RTNW	ND	B0	1 word
NSECTM1	Number of sections minus 1	NA	*RTNY, RTNK, RTNW, FMODE	ND	B0	1 word
NUMCTL	Maximum number of controllers in system	NA	*RTNY, RTND, RTNH, RTNI, RTNJ, RTNK, RTNM	ND	B0	1 word

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
NUMLNK	Number of links	NA	*RTNY, RTND, RTNH, RTNI, RTNJ	ND	B0	1 word
NUMSCT	Number of sections	NA	*RTNY, RTNH, RTNI, RTNJ, RTNK, RTNM, RTNW	ND	B0	1 word
NUMVAR	Number of variable (adjustable) intervals in a controller cycle	CONTR	*RTNY, RTNK, RTNS	ND	B0	1 byte/controller
NUMVED	Number of vehicle detectors	NA	*RTNY, RTNB, RTNE, RTNH, RTNJ, RTNK, RTNM, RTN5	ND	B0	1 word
N2	Zone 2 capacity correction factor	LINK	*RTNY, RTNC, RTND	VEH	B1	1 byte/link

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
PAR	Location of first entry in surveillance parameter table	NA	RTNJ	ND	NA	1 word
PLENGTH	Number of words in vehicle detector arrays	NA	*RTNA	ND	B0	1 word
SECCONSI	First controller index number for section in SECCONT	SECT	*RTNY, RTNH, RTNM, RTNW	ND	B0	1 byte/section
SECCONT	Controller number for each section pointed to by SECCONSI	SECC-ONSI	*RTNY, RTNH, RTNK, RTNM, RTNW	ND	B0	1 byte/SECCONSI entry
SECTLKN	Link numbers for each section pointed to by LINKSI	LINK	*RTNY, RTNI, RTNW	ND	B0	1 halfword/link
SEMMSK	Semiactuated controller mask array. Bit value of 0 indicates a semi	CONTR	*RTNY, RTNA1, RTNM, RTNT	ND	B0	1 bit/controller**
TT3	Maximum expected pulse duration for operating vehicles	NA	*RTNY, RTNE	SEC	B5	1 word
TT7	Maximum pulse reset time in a normally operating detector	NA	*RTNY, RTNE	SEC	B5	1 word

*Defining routine.

**Word interleaved.

Table 24. First Generation, UTCS, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
VARINT	Variable interval numbers INDEX = 4* (controller number) + (variable interval number)	INDEX	*RTNY, RTNK, RTNS, RTNT, RTNV LOADAT, PATGENT	ND	B0	4 bytes/controller
WD9	= 10VWD9 = 9 used to convert OACC to percent	NA	RTNH	ND	B0	1 word
WD900	= 10VWD900 = 900 used to convert QACC to 15-minute average	NA	RTNH	ND	B0	1 word
ZSI	First zone index number for each controller in CONTZN	CONTR	*RTNY, RTNH, RTNK, RTNM	ND	B0	1 byte/controller
Z2	Average number of stopped vehicles between second and third detectors when queue has extended up to and beyond third detector	NA	RTND	VEH	B1	1 word

*Defining routine.

Table 24. First Generation, UTCS, Static Arrays (Concluded)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
2NCWRD	Twice the number of words in a controller bit array minus 2	NA	*RTNY, RTNA1, RTNO	ND	B0	1 word
15TBYTE	Number of bytes in the 15-minute tape buffer	NA	*RTNY, RTNH	ND	B0	1 word
15TBSIZE	Number of words in the 15-minute tape buffer	NA	*RTNY, RTNH	ND	B0	1 word
15TINDEX	Fifteen-minute move tape buffer index	NA	RTNH	ND	B0	1 word

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
ABCEND	Interval number ending each controller phase; Byte 0, A-Phase Byte 1, B-Phase Byte 2, C-Phase Byte 3, not used	CONTR	*LLGTHF, PATGENT	ND	B0	4 bytes per controller
ABCSTRT	Interval number beginning each controller phase; Byte 0, A-Phase Byte 1, B-Phase Byte 2, C-Phase Byte 3, not used	CONTR	*LLGTHF, PATGENT	ND	B0	4 bytes per controller
AEXT	Percentage change in offset allowed on any given cycle in offset optimization logic of LOCAL	NA	*LOCAL	Percent	B0	1 word
CGRPN	Group to which each controller belongs	CONTR	*LINKD, RTNW	ND	B0	1 byte controller
CHNGTH	Decision threshold on change in delay above which newly computed pattern is implemented.	NA	*MACRO LSTSQS	Percent x 0.001	B0	1 word
CMITFT	Conversion factor to convert miles per hour to feet per second	NA	*LSTSQS	$\frac{\text{ft}}{\text{sec}} \Big\vert \frac{\text{mile}}{\text{hour}}$	B0	1 word

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CYCMAX	Upper limit on cycle length allowed to be generated by second generation optimization software	NA	*MACRO, RTSND	SECS	B0	1 word
CYCMIN	Lower limit on cycle length allowed to be generated by second generation optimization software	NA	*MACRO, RTSND	SECS	B0	1 word
GCONT	Controller numbers in each group	GROUP	*LINKD, GETINT, IRTN	ND	B0	1 byte/controller
GCONTSI	Starting index into GCONT array for each group	GROUP	*LINKD, GETINT, IRTN	ND	B0	1 byte/group
GRIPFC	Array defining which groups are associated with each group interface	GROUP INTERFACE	*MACRO	ND	B0	2 words/ interface
GRPNO	Group number of each group	GROUP	*LINKD	ND	B0	1 byte/group
GSECT	Section number to which each group belongs	GROUP	*MACRO	ND	B0	1 word/group
HEADWY	Headway between vehicles	NA	*LOCAL	SECS	B1	1 word
IALPH	Link importance factor for each second generation link. Range is 0 to 100.	LINK	*LLGTHF, GOMTRY	ND	B0	1 byte/link

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
IARG	Surveillance data control parameter = 0 use average link data for input to optimization algorithm = 1 use free flow link data for input to optimization algorithm	NA	*MACRO	ND	B0	1 word
ICGSI	Starting index into INCG array for each group	GROUP	*LINKD, IRTN	ND	B0	1 byte/group
INCG	Instrumented controllers in each group	ICGSI	*LINKD, IRTN	ND	B0	1 byte/controller
INFCR1	Cross reference giving group interfaces which are associated with each section interface	GROUP INTERFACE	*MACRO	ND	B0	1 word/group interface
INTN	Controller numbers along group interface belonging to base group	NGRPI	*INTFC	ND	B0	word/controller
INTNS	Controller numbers along section interface belonging to base section	NGRPIS	*INTFC	ND	B0	word/controller
INTN2	Controller numbers along group interface belonging to base group	NGRPI	*INTFC	ND	B0	word/controller

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
INTN2S	Controller numbers along section interface belonging to base section	NGRPIS	*INTFC	ND	B0	word/controller
IPRD	Surveillance data control parameter = 0 use average or free flow link data for input to optimization routine = 1 use predicted link data for input to optimization routine	NA	*MACRO, RTSND, LSTSQS, INTFC, TRNPAR	ND	B0	word
KV11	Converts reciprocal of pulse length to speed (mph)	NA	RTNFT	SEC/ MPH	B2	word
LENGTH	Distance from stop bar to stop bar for each second generation link	LINK	*LLGTHF, GOMTRY	FEET	B0	halfword/link
LFZE	Phasing code for each link. Indicates upstream and downstream controller phases at which traffic is released for each link	LINK	*LLGTHF, GOMTRY	ND	B0	byte/link

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
	Code / Upstream Controller Flow Phase / Downstream Controller Flow Phase					
	0 A A					
	1 A B					
	2 A C					
	3 B A					
	4 B B					
	5 B C					
	6 C A					
	7 C B					
	8 C C					
LINKN	Link numbers entering and departing INTN	GROUP INTERFACE	*INTFC	ND	B0	2 words per link pair
LINKNS	Link numbers entering and departing INTNS	SECTION INTERFACE	*INTFC	ND	B0	2 words per link pair
LL1	Occupancy correction factor	NA	*RTNFT	ND	B16	1 word
LMNO	Controller numbers associated with each link: Byte 0 second upstream controller number Byte 1 first upstream controller number Byte 2 first downstream controller number Byte 3 second downstream controller number	LINK	*LINKD, GOMTRY	ND	B0	1 word/link

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
LOCDAT	Array giving A-, B-, and C-Phase links associated with each critical intersection controller. See documentation of Routine LOCAL for structure of data layout	LOCFLG	*GOMTRY	ND	B0	1 byte/link (2 words per controller phase)
LOCFLG	Starting index into LOCDAT array for each critical intersection controller	CIC CON-TROLLER	*LINKD, RTNV, GOMTRY	ND	B0	1 byte per controller
NCIGRP	Number of controllers in each group	GROUP	*LINKD, GETINT, IRTN	ND	B0	1 byte per group
NGRIPF	Number of group interfaces	NA	*MACRO	ND	B0	1 word
NGRPI	Starting index into INTR and INTN2 arrays for each group interface	GROUP	*INTFC	ND	B0	1 word per group interface
NGRPIS	Starting index into INTNS and INTN2S arrays for each section interface	SECTION INTER-FACE	*INTFC	ND	B0	1 word per section interface
NGRPS	Number of groups in system	NA	*MACRO	ND	B0	1 word
NICIGP	Number of instrumented controllers in each group	GROUP	*LINKD, IRTN	ND	B0	1 byte per group

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
NINTF	Number of link pairs along each group interface	GROUP INTER-FACE	*INTFC	ND	B0	1 word per interface
NINTFS	Number of link pairs along each section interface	SECTION INTER-FACE	*INTFC	ND	B0	1 word per interface
NMAX	Maximum number of controllers allowed in one subnetwork	NA	*MACRO	ND	B0	1 word
N2GCL	Number of second generation links in each group	GROUP	*LINKD	ND	B0	1 byte per group
OCCW	Weighting factor on occupancy to be used in determining cycle length and split. Units are vehicles per hour at 100 percent occupancy	NA	*MACRO	See Description	B0	1 word
OCTH	Threshold on occupancy in computing optimum cycle length and split. If occupancy exceeds threshold, weighted occupancy is included in computations	NA	*MACRO	Percent	B0	1 word

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
OPTIME	MACRO loop scheduling frequency	NA	*RTNY, RTNFT, RTNA	Minutes	B0	1 word
OPTIME	Time between successive Traffic Responsive Pattern Implementations	NA	*MACRO, INTFC	Minutes	B0	1 word
SATHLW	Lower threshold on total green time required in cycle divided by total green time available in cycle below which time of day split is implemented for instrumented controllers. MACRO loop threshold	NA	*MACRO, RTSND	Percent x 0.001	B0	1 word
SATHRH	Upper threshold on total green time required in cycle divided by total green time available in cycle above which time of day split is implemented for instrumented controllers. Theory is that when demand exceeds this threshold, the queues have been extended beyond the capability of the system to accurately measure demand. MACRO loop threshold	NA	*MACRO, RTSND	Percent x 0.001	B0	1 word

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
SATTHL	Same as SATHLW except applies to MICRO loop control	NA	LOCAL	Percent	B0	1 word
SATTHL	Same as SATHRH except applies to MICRO loop control	NA	LOCAL	Percent	B0	1 word
SECDFF	Factor used to multiply speed in computing idealized offsets to account for secondary flow queue buildup	NA	*MACRO, LSTSQS, INTFC	ND	B0	1 word
SECIFC	Array which indicates sections associated with each section interface	ND	*MACRO	ND	B0	2 words per section interface
SPLCHT	Ratio green demand to green time available on any approach above which MACRO loop split computations at instrumented intersections are attempted	NA	*MACRO, RTSND	Percent x 0.001	B0	1 word
SPLCHT	Same as SPLCHT except applies to MICRO loop control	NA	LOCAL	Percent		1 word
TBSIZE	Size of RTN5 tape buffer in words	NA	*RTN5, LDMOE5	Words	B0	1 word

*Defining routine.

Table 25. Second Generation, TANSTP, Static Arrays (Concluded)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
TOL	Tolerance on cycle length between two neighboring groups below which groups are combined into one subnetwork	NA	*MACRO, RTSND	Seconds	B0	1 word
1GLNK	First generation link number associated with each second generation link	SECOND GENERATION LINK	*LINKD, GOMTRY, LKCROS	ND	B0	1 byte per link
2GCLSI	Second generation link starting index for each group	GROUP	*LINKD	ND	B0	1 byte per group

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays

Mnemonic	Definition	Index	Defining & Reference Routine / Subroutine	Units	Scale	Format
AOG	Corrected A-Phase green return array. Each bit represents the status of a particular A-Phase green return signal a bit-value of 1 = presence	CONTR	*RTNO, RTNA1, RTNP, RTNX,	ND	B0	1 bit/controller**
APO	Advance pulse output data	CONTR	*RTNA, RTNA1, RTNY	ND	B0	1 bit/controller**
ADVINH	Advance pulse inhibit flag	CONTR	*RTNY, RTNP, RTNT	ND	B0	1 bit/controller**
AGR	A-Phase green return	CONTR	*RTNA	ND	B0	1 bit/controller**
AHCC	Smoothed occupancy	LINK	*RTNY, RTNF, RTNJ, RTNK, RTNW, LDMOE	ND	B15	1 halfword/link
AHHIST	Occupancy history for the pattern read in RTNW	LINK	*RTNY, RTNF, RTNJ, RTNK, RTNW, RTNFT	ND	B15	1 halfword/link

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
AUTOF	Automatic control desired flag	CONTR	*RTNY, RTNM, RTNT, RTNW, RTNX	ND	B0	1 bit/controller
AVSP	Average speed temporary storage	NA	*RTNY, RTNH, RTNI	MPH	B5	1 word
BYTCNT	Number of bytes in CRT buffer. BYTCNT(0) is for CRT 2; BYTCNT(1) is for CRT 1	CRT NO.	*CRTOUT, RTNK	ND	B0	1 word/CRT
CC1	Link car count	LINK	*RTNY, RTNC, RTND	VEH	B0	1 byte/link
CC2	Detector car count	DET	*RTNY, RTNC, RTND, RTNE	VEH	B0	1 byte/detector

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CHGTABLE	Change Table: 5 words for each entry CHGTABLE(I) = TIME CHGTABLE(I+1) = TYPE CHGTABLE(I+2) = NUMBER CHGTABLE(I+3) = CHANGE CHGTABLE(I+4) = NUMBER if manual pattern change, EXTENSION if manual extension was inserted, zero for all other changes	IBL	RTNH	ND	B0	5 words/entry
CICBYP	CIC bypass flags set during transitional cycles	CONTR	*RTNY, RTNT, RTNV	ND	B0	1 bit/controller**
CICCNT	Count for active CIC's	NA	*RTNK, RTNH	ND	B0	1 word
CICEF	CIC entry flags. Set when all conditions are met enabling controller to operate as a CIC	CONTR	*RTNY, RTND, RTNH, RTNJ, RTNK, RTNM, RTNP, RTNR, RTNV	ND	B0	1 bit/controller**

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CICSF	CIC selected flag. Set when operator designates controller is CIC	CONTR	*RTNY, RTNH, RTNJ, RTNK, RTNM, RTNP, RTNQ, RTNR, RTNV	ND	B0	1 bit/controller**
CICTF	CID threshold flag. Set when CIC split control is being imposed	CONTR	*RTNY, RTND, RTNK, RTNP, RTNR, RTNV	ND	B0	1 bit/controller**
CICTODF	CIC time-of-day flags used to inhibit CIC as a function of TOD	CONTR	*RTNY, RTNK, RTNM, RTNV, RTNW	ND	B0	1 bit/controller**
CIRCR	Controller interval numbers to be reduced	MOE-PER	*RTNY, RTND	ND	B0	1 byte/MOE period

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CMALF	Controller malfunction flags	CONTR	*RTNY, RTNH, RTNJ, RTNK, RTNM, RTNP, RTNQ, RTNX, RTNW	ND	B0	1 bit/controller**
CMMALF	Detector communications malfunction flags.	DET	*RTNY, RTNE, RTNH, RTNJ, RTNK, RTNM, RTNQ	ND	B0	1 bit/detector
CNTDR	Countdown registers	CONTR	*RTNY, RTND, RTNS, RTNT, RTNU, RTNV, RTNX	SEC	B1	1 byte/controller
COMKQ1	1 - KQ1	NA	*RTNY, RTND	ND	B31	1 word

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
COMKQ2	1 - KQ2	NA	*RTNY, RTND	ND	B31	1 word
COMKV0	1 - KV0	NA	*RTNY, RTND	ND	B31	1 word
COMLENG	Number of communication failures to be logged from COMTAB	NA	*RTNY, RTNE, RTNK	ND	B0	1 word
COMTAB	Unit number of each failed communications	NA	*RTNY, RTNE, RTNK	ND	B0	1 halfword/unit
CONDIS	Controller display code	NA	*RTNY, RTNJ, RTNM	ND	NA	1 word
CONLENG	Number of controller failures to be logged from CONTAB	NA	*RTNY, RTNK, RTNP, RTNX	ND	B0	1 word
CONTAB	Unit number of each failed controller	NA	*RTNY, RTNK, RTNP, RTNX	ND	B0	1 halfword/unit

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CONTNM	Controller number being processed	NA	*RTNY, RTNG, RTNS, RTNT, RTNV	ND	B0	1 word
CPAOG	Computed start of A-Phase green	CONTR	*RTNY, RTNP, RTNT, RTNA1	ND	B0	1 bit/controller**
CPANIN	Control panel input each bit represents the status of a particular control panel input signal a bit-value of 0 = PRESENCE	NA	*RTNA, RTNM	ND	NA	1 bit/input
CPANOU	Control panel output command each bit represents the status of a particular control panel output signal a bit-value of 1 = PRESENCE	NA	*RTNA	ND	NA	1 bit/output
CPANOUT	Control panel output data	NA	*RTNA, RTNK, RTNL, RTNM, KFMT	NA	NA	1 word

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CPFLG	Flag set by CPU 1 upon initiating the programmed computer evaluation mode	NA	*RTNL, RTNA	ND	B0	1 word
CRTAC1	Flag set when buffer for CRT 1 is to be output	NA	*RTNK, CRTOUT	ND	B0	1 word
CRTAC2	Flag set when buffer for CRT 2 is to be output	NA	*RTNK, CRTOUT	ND	B0	1 word
CRTBF1	Output buffer for CRT 1	CHAR-ACTER	*RTNY, RTNA, RTNK, RTNQ, CRTOUT	NA	B0	1 byte/character
CRTBF2	Output buffer for CRT2	CHAR-ACTER	*RTNY, RTNA, RTNK, RTNQ, CRTOUT	NA	B0	1 byte/character
CRTDC1	CRT 1 display code. Code is: 1=FAIL; 2=SYS; 3=CON; 4=DET; 5=INT; 6=BPS SYS; 7=BPS INT	NA	*RTNY, RTNK, RTNL, RTNM	ND	B0	1 word
CRTDC2	CRT 2 display code. Code is: 1=FAIL; 2=SYS; 3=CON; 4=DET; 5=INT; 6=BPS SYS; 7=BPS INT	NA	*RTNY, RTNK, RTNL	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CRTERR2	Error flag for CRT term's Word 0 is for CRT 2 Word 1 is for CRT 1	NA	*CRTOUT, RTNK, RTNQ	ND	B0	1 word/CRT
CRTFL1	Flag indicating operator selection of CRT 1 for display	NA	*RTNY, RTNA, RTNK, RTNL, RTNM, CRTOUT	ND	B0	1 word
CRTFL2	Flag indicating operator selection of CRT 2 for display	NA	*RTNY, RTNA, RTNK, RTNL, RTNM	ND	B0	1 word
CRTNUM	Flag indicating CRT to which data is sent. Code is: 0 = CRT 2; 1 = CRT 1	NA	*RTNK, CRTOUT	ND	B0	1 word
CUROFF	Current offset (Interval 1)	CONTR	*RTNY, RTNK, RTNU, RTNX	SEC	B1	1 byte/controller

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
CYCLC	Cycle length codes. Code is: 0=40 SEC, 1=50 SEC, 2=60 SEC, 3=70 SEC, 4=80 SEC, 5=90 SEC	CONTR	*RTNY, RTND, RTNE, RTNH, RTNK, RTNM, RTNS, RTNU, UNPAK, GOMTRY, GETCYCLC, RTNX, RTNT	ND	B0	1 byte/controller
D	Block length	LINK	*RTNY, RTND, RTNH, RTNI	FT	B0	1 halfword/link
DACC	Fifteen-minute accumulation of delay	LINK	*RTNY, RTND, RTNH	SEC	B1	1 halfword/link

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine / Subroutine	Units	Scale	Format
DATE	Current date in EBCDIC. In DATE(0), Bytes 0, 1 and 2 contain a 3 digit symbol for the month and Byte 3 contains a blank. In DATE(1), Byte 0 and 1 contain the day, Byte 2 contains a blank and Byte 3 contains the tens digit of the year. In DATE(2), Byte 0 contains the units digit of the year.	NA	*RTNH, RTNI, RTNJ, RTNK, RTNM, RTNQ, RTNY	SEE DESCR	NA	3 words
DAY	Day column of failure status report	NA	RTNH	ND	B0	1 word/entry
DAYOWK	Day of week. Codes are: 1=SAT; 2=SUN; 3=MON; 4=TUE; 5=WED; 6=THU; 7=FRI	NA	*RTNY, RTNI, RTNW	ND	B0	1 word
DBP	Detector buffer pointer	NA	*RTNY, RTNA, RTNB	ND	B0	1 word
DCF	Disc call flag. Set if a pattern has been read in for the present section	NA	*RTNY, RTNW	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
DELAY	Average delay per vehicle during the past 15 minutes	LINK	*RTNY, RTNH, RTNJ, RTNK	SEC	B4	1 halfword/link
DELDAY	Accumulated delay in a day	LINK	*RTNY, RTNH, RTNI	SEC	B1	1 halfword/link
DETLENG	Number of detector failures to be logged from DETTAB	NA	*RTNY, RTNE, RTNK	ND	B0	1 word
DETTAB	Unit number of each failed detector	NA	*RTNY, RTNE, RTNK	ND	B0	1 halfword/unit
DFAIL	Set when a detector has failed on a link	LINK	*RTNY, RTNE, RTNH, RTNI, RTNJ, RTNK, LNKFL, PREDICT, RTNFT	ND	B0	1 byte/link

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
DIS	Detector in standby flag. Set when a detector is placed into standby by operator	DET	*RTNY, RTNE, RTNH, RTNK, RTNM, RTNQ, RTN5	ND	B0	1 bit/detector
DM	Corrected car counts used in smoothing logic	NA	RTNC	VEH	B3	1 word
DMALF	Detector failure flags. Set when detector electronics failure is indicated or when detector gives spurious excessive counts	DET	*RTNY, RTNE, RTNH, RTNJ, RTNK, RTNM, RTNQ, RTN5	ND	B0	1 bit/detector
DMODE	Desired Mode. Code is: 0=STANDBY; 1=MANUAL; 2=TIME OF DAY; 3=TRAFFIC RESPONSIVE	SECT	*RTNY, RTNH, RTNJ, RTNK	ND	B0	1 byte/section
DOF	Detector obstruction flag	DET	*RTNY, RTNC, RTNE, RTN5	ND	B0	1 bit/detector

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
DOFF	Delta offset time	NA	RTNS	SEC	B1	1 word
DOW	Day of week unless holiday flag is set for either a Sat or Sun code. Code is: 1=SAT; 2=SUN; 3=MON; 4=TUE; 5=WED; 6=THUR; 7=FRI	NA	*RTNY, RTNW	ND	B0	1 word
DPATNUM	Desired pattern number	SECT	*RTNY, RTNM, RTNW	ND	B0	1 byte/section
DSGSF	Downstream green signal flags	LINK	*RTNY, RTNC, RTND	ND	B0	1 bit/link
EBCMODE	Modes in EBCDIC format 'STBY', 'MAN', 'TOD', 'TRSP'	MODE CODE	RTNH	ND	B0	1 word
EMAST	Priority of RTNE in mast	NA	*RTNY, RTNA, RTNN	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
ENTFLG	Failure entry flags. Word 0 = -1 if no controller failures to log; = 0 if some to log. Words 1-4 = 0 if no detector, communication, failures, respectively, to log; ≠ 0 if some to log	SEE DESCR	*RTNY, RTNE, RTNK, RTNP, RTNX	ND	B0	5 words
ENTRYINT	Entry interval. Duration of controller intervals. Index = 16* (controller number) + (interval number)	INDEX	*RTNY, RTNK, RTNS, RTNT, RTNV, RTNW, RTNX, SMINT, LOADAT, PATGENT	SEC	B1	16 bytes/ controller

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
ENTRYOFF	Offset as read in for each pattern (Interval 1)	CONTR	*RTNY, RTNK, RTNM, RTNS, RTNT, RTNW, RTNX, LOADAT, PATGENT, GETEOFF, PUTEOFF, RTNH	SEC	B1	1 byte/controller
EODC	End of day count indicates the number of end of day reports	NA	*RTNI, RTNQ, RTNY	ND	B0	1 word
EPSILON	Temporary storage	NA	RTNX	ND	NA	1 word
EPSOF	Difference between actual and working offset	NA	RTNU	SEC	B1	1 word
F	Comparison flag (=0 turns on link arrows for links having parameter values greater than threshold; turns on those having values less than threshold)	NA	RTNJ	ND	NA	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
FAIL	Fifteen-minute periods that detectors or communications have failed	ZONE	*RTNY, RTNH, RTNI	ND	B0	1 byte/zone
FAILTAB	Failure table. Format is described in RTNK spec	SEE DESCR	*RTNY, RTNH, RTNK, RTNM, KFMT, RTNQ	SEE DESCR	NA	1 word
FAILTB	Equated to FAILTAB					
FFS	Free flow link speed	TTSA-SI	*RTNY RTND	FT/ SEC	B0	1 halfword
FLERFL	Failure ack/rep error flag	NA	*RTNY, RTNK, RTNM, KFMT	ND	B0	1 word
FMC	Fifteen-minute counter	NA	*RTNY, RTNA	SEC	B0	1 word
FMMARK	Fifteen-minute mark. Indicates that RTNW should process all sections	NA	*RTNY, RTNA, RTNW	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
FMSECF	Fifteen-minute section flag	NA	*RTNY, RTNA, RTNM, RTNW	ND	B0	1 word
FMSYSF	Fifteen-minute system flag	NA	*RTNY, RTNA, RTNM, RTNW	ND	B0	1 word
FMTCHG	CRT formatting change parameter	NA	*RTNM, RTNK, KFMT	ND	B0	1 word
FMTNUM	CRT formatting number parameter	NA	*RTNM, RTNK, KFMT	ND	B0	1 word
FMTODC	Fifteen-minute time of day count. Reference is midnight. Incremented every 15 minutes MOD 96	NA	*RTNY, RTNA, RTNH, RTNI, RTNW	ND	B0	1 word
FMTTYP	CRT formatting type parameter	NA	*RTNM, RTNK, KFMT	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
FPPF	First pass parameter flag. Code is: 0=W has been entered at least two times since start-up; 1=W has been entered one time since start-up; 2=first time RTNW has been entered since start-up	NA	*RTNY, RTNW	ND	B0	1 word
FSETCF	Offset change flags. Set when there is a change in offset due to operator request, or corrective action is in progress	CONTR	*RTNY, RTNG, RTNM, RTNT, RTNU, RTNV, RTNX	ND	B0	1 bit/controller**
FSETF	Equated to FSETCF					
FSTFLG	First start flag	NA	*RTNY, RTNM	ND	B0	1 word
FTAB	Byte index points to first byte of next available entry in FAILTAB	NA	*RTNK, RTNQ, RTNY, KFMT	ND	B0	1 word
FTABWRD	Word index points to second word of next available entry in FAILTAB	NA	*RTNK, KFMT, RTNQ	ND	B0	1 word

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
GA	Calculated duration of A-Phase green plus amber	NA	RTNS	SEC	B1	1 word
GA1	A-Phase green return Input Buffer Number 1	CONTR	RTNO	ND	NA	1 bit/controller**
GA2	A-Phase green return Input Buffer Number 2	CONTR	RTNO	ND	NA	1 bit/controller**
GABBCABC	Entry (Time-of-Day) controller phasing data. Contents depend controller type as follows: Controller Type — Contents K3PH = 0: Byte 0, time available for A- and B-Phase Green; Byte 1-3, not used K3PH = 1: Byte 0, time available for A- and B-Phases Green; Byte 1, time available for B- and C-Phase Green	CONTR	*RTNY, RTNM, RTNS, RTNV, RTNW, PATGENT, LPHASV	SEC (EACH BYTE)	B1	4 bytes/ controller

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
	Controller Type / Contents K3PH = 1: Byte 2, time available for A-, B-, and C-Phase Green K3PH =2: Byte 0, time available for A- and B-Phase Green; Byte 1, time available for B- and C-Phase Green; Byte 2 and 3, not used K3PH = 3: Byte 0, time available for A- and B-Phase Green; Byte 1, time available for A- and C-Phase Green					
GABCP	Time for A-, B-, and C-Phase green plus amber during a transition cycle	NA	RTNS	SEC	B1	1 word
GABP	Time for A- and B-Phase green plus amber during a transition cycle	NA	RTNS	SEC	B1	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
GAI	A-Phase green return input each bit represents the status of a particular A-Phase green return signal a bit-value of 0 = PRESENCE	CONTR	*RTNA, RTNN	ND	B0	1 bit/controller**
GAMIN	Temporary storage for GMINABC4 value	NA	RTNM	SEC	B1	1 word
GAP	Present gap length on detector being processed	NA	RTNE	SEC	B5	1 word
GAP1	A-Phase green return input buffer pointer	NA	RTNO	ND	NA	1 word
GATEMP	Control panel thumbwheel value of A-Phase green (binary)	NA	RTNM	SEC	B1	1 word
GAW	Temporary storage for controller interval duration	NA	RTNM	SEC	B1	1 word
GAWTEMP	Operator adjusted A-Phase green	CONTR	*RTNY, RTNM, RTNT	SEC	B1	1 byte/controller
GB	Calculated duration of B-Phase green plus amber	NA	RTNS	SEC	B1	1 word
GBMIN	Temporary storage for GMINABC4 value	NA	RTNM	SEC	B1	1 word

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
GBTEMP	Control panel thumbwheel value of B-Phase green (binary)	NA	RTNM	SEC	B1	1 word
GBW	Temporary storage for controller interval duration	NA	RTNM, RTNV	SEC	B1	1 word
GBWTEMP	Operator adjusted B-Phase green	CONTR	*RTNY, RTNM, RTNT	SEC	B1	1 byte/controller
GC	Calculated duration of C-Phase green plus amber	NA	RTNS	SEC	B1	1 word
GMINABC4	GAMIN (Byte 0), GBMIN (Byte 1) and GCMIN (Byte 2) are the minimum allowable durations of A-Phase, B-Phase, and C-Phase green, respectively. G4MIN (Byte 3) is the minimum allowable minor adjustable interval duration	CONTR	*RTNY, RTNK, RTNS, RTNW, LOADAT, UNPAK, LPHASV	SEC	B1	4 bytes/ controller
GT	Temporary storage used in green return calculations	NA	RTNO	ND	NA	1 bit/controller
HBUFLNG	Length of history buffer: NPATBYT4 + 256*NRESPAT	NA	*RTNY, RTNW	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
HMAST	Priority of RTNH in mast	NA	*RTNY, RTNA, RTNH, RTNM	ND	B0	1 word
HOLD	Flgas indicating if controller is under computer control. Each bit represents the status of a particular controller on-line signal a bit-value of 1 = PRESENCE	CONTR	*RTNA, RTNA1, RTNH, RTNI, RTNJ, RTNK, RTNM, RTNP, RTNQ, RTNR, RTNT, RTNV, RTNX	ND	B0	1 bit/controller**
HOLDTEMP	On-line fixed time controller indication	NA	RTNT	ND	NA	1 bit/controller
HOLIDF	Holiday flag. Code is: 0 = NO HOLIDAY 1 = SATURDAY type holiday 2 = SUNDAY type holiday	NA	*RTNY, RTNI, RTNW, PREDICT	ND	B0	1 word

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
HOLIDTBL	Holiday table listing date and code type of each holiday. In HOLIDTBL(0), Bytes 0, 1, and 2 contain a 3 digit code for the month, and Byte 3 contains a blank. In HOLIDTBL(1), Bytes 0 and 1 contain the day, Byte 2 contains a blank and Byte 3 contains the tens digit of the year. In HOLIDTBL(2), Byte 0 contains the units of the year, Byte 1 contains a blank, and Byte 2 contains the code type. The code is: 1 = SAT type; 2 = SUN type. Storage is provided for 32 holidays	SEE DESCR	*RTNY, RTNI, RTNQ	ND	B0	3 words
HSARR	Half-second array contains A-Phase green input (16 words), radio-synch (2 words) and hold (8 words). Data is interleaved with commands.	NA	*RTNA	ND	B0	1 bit/status**

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
I	Do loop word index	NA	*RTNY, RTNA1, RTNG, RTNP, RTNR, RTNS, RTNT, RTNU, RTNV, RTNX	ND	B0	1 word
IADVP	Interval advance pulse temporary storage	CONTR	*RTNY, RTNA1, RTNP, RTNT, RTNX	ND	B0	1 bit/controller**
IB	Cycle number index	NA	*RTNE, RTNF	ND	B0	1 word
IBL	Change table index	NA	RTNH	ND	B0	1 word
IBRANIND	Branch index points to the starting address of RTNI	NA	RTNI	ND	B0	1 word
IC	Controller interval number to be reduced	NA	RTND	ND	B0	1 word
IL	Link number	NA	*RTNE, RTNF	ND	B0	1 word

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
IM	MOE period number to be reduced	NA	RTND	ND	B0	1 word
IMAST	Priority of RTNI in mast	NA	*RTNY, RTNA, RTNH, RTNI	ND	B0	1 word
INHIBITE	Inhibit Routine E flag	NA	*RTNY, RTNE, RTNH	ND	B0	1 word
INHT	Number of intersections that are in inhibited mode	NA	RTNH	ND	B0	1 word
INT1SF	Interval 1 start flag	CONTR	*RTNY, RTNA1, RTNT, RTNV	ND	B0	1 bit/controller**
INT1SHF1	Interval 1 start-hold flag 1. Set at the beginning of A-Phase green and reset when the MOE1 period is loaded	CONTR	*RTNY, RTND, RTNT	ND	B0	1 bit/controller**

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
INTDUR	Interval duration array INDEX = controller interval number + 16* (controller number)	INDEX	*RTNY, RTNV, RTNK, RTNS, RTNT, RTNV, RTNX	SEC	B1	16 bytes/ controller
INTNUM	Interval number	CONTR	*RTNY, RTNT, RTNX	ND	B0	1 byte/controller
INTNUML	Previous interval number	CONTR	*RTNY, RTNP, RTNT	ND	B0	1 byte/controller
INTOUT	Advance pulse counter	CONTR	*RTNY, RTNP	ND	B0	1 byte/controller
INTRNUM	Intersection number from RTNM. Word 0 is for CRT 2 and Word 1 is for CRT 1	NA	*RTNR, RTNM	ND	B0	2 words
I1	Current MOE1 period number	CONTR	*RTNY, RTND	ND	B0	1 halfword/ controller
I2	Current MOE2 period number	CONTR	*RTNY, RTND	ND	B0	1 halfword/ controller

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine / Subroutine	Units	Scale	Format
JCOUNT	Pass counter for J (6 sec)	NA	*RTNA, RTNJ	ND	B0	1 word
JJ	Section number - 1	NA	*RTNY, RTNM, RTNW	ND	B0	1 word
JMAST	Priority of RTNJ in mast	NA	*RTNY, RTNA, RTNM	ND	B0	1 word
K	Index indicating the first controller number of the current group of 32 being processed by the controller do-loop routines	NA	*RTNY, RTNA1, RTNP, RTNR, RTNS, RTNT, RTNU, RTNV, RTNX	ND	B0	1 word
KCNT	Time till next CRT update	NA	*RTNA, RTNK	SEC	B0	1 word
KCSF	Cycle K start flags. Set when kth cycle starts	CYCLC	*RTNN, RTNE	ND	B0	1 bit/cycle length

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
KCTC	Kth cycle time counters contain time remaining in cycle	CYCLC	*RTNN, RTNS, RTNU, RTNX, ENTCYCL, GETCYCLC	SEC	B1	1 word/cycle length
KCTF	Kth cycle timing flag. Flag Flag set for each cycle to indicate that synch has been obtained with the master synch pulses	CYCLC	*RTNN, RTNK	ND	B0	1 bit/cycle length
KMAST	Priority of RTNK in mast	NA	*RTNY, RTNA, RTNE, RTNK, RTNL, RTNM, RTNP, RTNX	ND	B0	1 word
KMCSP	Kth master controller synchronization pulses	CYCLC	*RTNA, RTNN	ND	B0	1 bit/cycle length
LAG	Lagging controllers	NA	RTNP	ND	NA	1 bit
LEAD	Leading controllers	NA	RTNP	ND	NA	1 bit

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
LNKSPEED	Link speed. Total accumulation per cycle of raw speeds of all detectors on link	NA	*RTNY, RTNE, RTNF	SEC^{-1}	B4	1 word
LPBUF	Line printer buffer	NA	*RTNH, RTNI	ND	NA	1 word
MARR	Map array consisting of output code words interleaved with data array MARR1	NA	*RTNA, RTNJ, RTNY	ND	B0	1 bit/status**
MARR1	Label equated to MARR+1					
MAST	Master activity scan table. Bits 0, 1, and 2 contain the activity, service and control bits, respectively. Bits 15 - 31 contain the address of the Level 3 RTN. The words are arranged with the highest priority at the top of the table. The priority order is: M, J, K, E, H, W, Q, I, L	PRIORITY	*RTNY, RTNA, RTNE, RTNH, RTNI, RTNJ, RTNK, RTNL, RTNM, RTNN, RTNP, RTNW, RTNX	ND	SEE DESCR	1 word

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
MCDF	Mode change desired flag	NA	*RTNY, RTNM, RTNW	ND	B0	1 bit/section
MI	Number of transition cycles remaining	CONTR	*RTNY, RTND, RTNK, RTNM, RTNP, RTNQ, RTNR, RTNS, RTNT	ND	B0	1 byte/controller
MMAST	Priority of RTNM in mast	NA	*RTNY, RTNA	ND	B0	1 word
MODE$	Current operating mode code is: 0 = STANDBY; 1 = MANUAL; 2 = TIME OF DAY; 3 = TRAFFIC RESPONSIVE	SECT	*RTNY, RTNH, RTNK, RTNM, RTNW, FMODE	ND	B0	1 byte/section

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
MOE1F	MOE inactive flags giving failure status. Code equal to '0' if no failure was present for more than one cycle, equal to '1' if failure was corrected for one cycle only, equal to '2' if there is a failure present	LINK	*RTNY, RTNC, RTND, RTNE, RTNH, RTNI, RTNK, RTNW, RTNV	ND	B0	1 byte/link
MOETMP	Temporary storage for duration of adjustable MOE1 period	NA	RTNG	SEC	B1	1 word
MOETP2	Temporary storage for duration of adjustable MOE2 period	NA	RTNG	SEC	B1	1 word
MOE1CLK	Clocks used to countdown the MOE1 periods	CONTR	*RTNY, RTND	ND	B1	1 byte/controller
MOE2CLK	Clocks used to countdown the MOE2 periods	CONTR	*RTNY,	ND	B1	1 byte/controller
MTHLD	Integer value of parameter threshold	NA	RTNJ	ND	NA	1 word
MVPLUSKO	Measured volume plus weighted occupancy	NA	RTNH	VPH	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
MONTH	Month column of failure status report	NA	RTNH	ND	B0	1 word/entry
MOVTAB	Shutdown/startup table containing the arrays to be saved/restored	NA	*RTNY, RTNQ	ND	B0	1 halfword/array
MPOKEY	Must print out key. Set when CHNGTBL is full	NA	*RTNH, RTNA	ND	B0	1 word
NEXTPAGE	Set when a report takes more than a single page	NA	RTNH	ND	B0	1 word
NOFAIL	Number of failures in FAILTB	NA	*RTNY, RTNK, RTNH, RTNM	NA	B0	1 word
NHOLIDAY	Number of holidays in HOLIDTBL	NA	*RTNY, RTNI, RTNQ	ND	B0	1 word
NI	Number of detectors on link being processed	NA	*RTNY, RTNE, RTNF, RTNH	ND	B0	1 word
NOTINTOL	Number of links that are not in tolerance	NA	RTNH	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
NREPORT	Number of 15-minute reports	NA	*RTNH, RTNI	ND	B0	1 word
NSACC	Fifteen-minute accumulation of number of stops	LINK	*RTNY, RTND, RTNH	ND	B0	1 byte/link
NSDAY	Accumulated stops in a day	LINK	*RTNY, RTNH, RTNI	VEH	B0	1 halfword/link
NSTOPS	Smoothed number of stops per cycle	LINK	*RTNY, RTND, RTNJ, RTNK	ND	B0	1 byte/link
NSUTC	Start-up time counter	NA	*RTNN	SEC	B1	1 word
NUMMOV	Shutdown/startup table containing the number of words to save/restore for the arrays listed in MOVTAB	NA	*RTNY, RTNQ	ND	B0	1 halfword/array
OACC	Fifteen-minute accumulation of total lane occupancy	LINK	*RTNY, RTNF, RTNH	SEC	B5	1 halfword/link
ODAY	Accumulated percent occupancy in a day	LINK	*RTNY, RTNH, RTNI	PER	B2	1 halfword/link

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
OPC	Temporary storage of percent occupancy	NA	*RTNY, RTNH, RTNI	PER	B5	1 word
P	Corrected pulse duration used as array INDEX	NA	RTNC	ND	B0	1 word
PAGEKEY	Set when a report takes more than a single page	NA	RTNH	ND	B0	1 word
PAGNUM	Page number for 15-minute and end of day reports	NA	*RTNH, RTNI	ND	B0	1 word
PARCOD	Flow data parameter code	NA	*RTNY, RTNM, RTNJ	ND	B0	1 word
PATNUM	Present pattern number	SECT	*RTNY, RTNH, RTNK, RTNW	ND	B0	1 byte/section
PCOT	Temporary storage of link occupancy	NA	RTNF	ND	B7	1 word
PDOF	Possible detector obstruction flag	DET	RTNE	ND	NA	1 bit/detector
PLS	Present pulse length of detector being processed	NA	RTNE	SEC	B5	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine / Subroutine	Units	Scale	Format
PREVDATE	Previous date	MON, DAY, YR	*RTNI, RTNH	ND	EBC-DIC	1 word
PRTFLG	Fifteen-minute report suppress flag. = 0, PRINTOUT; = 1, NO PRINTOUT	NA	*RTNM	ND	B0	1 word
PTN	Pattern read in from RAD for present section	NA	*RTNY, RTNW	NA	NA	word
PW	Corrected pulse duration used in smoothing logic	NA	RTNC	SEC	B5	1 word
PWL	Lower stops threshold. When PW2 exceeds PWL and other criteria (see RTND flow chart) are present, probability is high that stopped queue extends beyond second detector	NA	*RTNY, RTND	SEC	B5	1 word
PW0	Upper threshold for vehicle stops. When PW2 or PW3 exceeds this threshold, probability is high that last vehicle across detector has or will stop immediately downstream of detector	NA	*RTNY, RTND	SEC	B5	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
PW2	Smoothed pulse width of second detector	LINK	*RTNY, RTNC, RTND	SEC	B5	1 halfword/link
PW3	Smoothed pulse width of third detector	LINK	*RTNY, RTNC, RTND	SEC	B5	1 halfword/link
P1	Vehicle detector Array 1 each bit represents the status of a particular vehicle detector signal a bit-value of 0 = PRESENCE	DET	*RTNA, RTNB	ND	B0	1 bit/detector**
P2	Vehicle detector Array 2 each bit represents the status of a particular vehicle detector signal a bit-value of 0 = PRESENCE	DET	*RTNA, RTNB	ND	B0	1 bit/detector**
P3	Vehicle detector Array 3 each bit represents the status of a particular vehicle detector signal a bit-value of 0 = PRESENCE	DET	*RTNA, RTNB	ND	B0	1 bit/detector**

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
P4	Vehicle detector Array 4 each bit represents the status of a particular vehicle detector signal a bit-value of 0 = PRESENCE	DET	*RTNA, RTNB	ND	B0	1 bit/detector**
Q	Temporary storage for queue	NA	RTND	VEH	B3	1 word
QACC	Fifteen-minute accumulation of queue	LINK	*RTNY, RTND, RTNH	VEH	B3	1 byte/link
QBAR	Smoothed queue	LINK	*RTNY, RTND, RTNJ, RTNK, LDMOE	VEH	B0	1 byte/link
QCIC	Temporary storage for QCICT value	NA	RTND, RTNV	VEH	B2	1 word
QCICT	Smoothed stopped queue at beginning of link green	LINK	*RTNY, RTND, RTNV	VEH	B2	1 byte/link
QCNT	Timing count for scheduling RTNQ	NA	*RTNQ, RTNA	SEC	B0	1 word

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
QDAY	Accumulated queues per cycle in a day	LINK	*RTNY, RTNH, RTNI	VEH	B5	1 halfword/link
QG	Raw queues at start of green phase	LINK	*RTNY, RTND	VEH	B3	1 byte/link
QM	Number of vehicles in queue	TTSA	RTNC	VEH	B2	1 word
QMAST	Priority of RTNQ in mast	NA	*RTNY, RTNA, RTNI, RTNL, RTNQ, RTNW	ND	B0	1 word
QPT	Temporary storage of queue per cycle	NA	*RTNY, RTNH, RTNI	VEH	B5	1 word
QR	Temporary storage of raw queue at start of red	NA	RTND	VEH	B3	1 word
QRD	Raw queue at start of red	LINK	*RTNY, RTND	VEH	B0	1 byte/link

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
QVOK	Flags indicating failure status. Code is '0' whenever any component on link has failed; '1' for one full action period (full green or red period) following correction of failure; '2' thereafter	LINK	*RTNY, RTND, RTNM, RTNV	ND	B0	1 byte/link
Q1	Smoothed queue in zone	LINK	*RTNY, RTNC, RTND	VEH	B3	1 byte/link
RAWLNKC	Raw link count. Total count per cycle of all detectors on link being processed	NA	*RTNY, RTNE, RTNF	VEH	B0	1 word
RAWLNKO	Raw link occupancy. Total accumulation per cycle of raw occupancy of all detectors on link being processed	NA	*RTNY, RTNE, RTNF	SEC	B1	1 word
RAWOCC	Raw occupancy	DET	*RTNY, RTNC, RTNE	SEC	B5	1 halfword/ detector
RAWSP	Raw speed	DET	*RTNY, RTNC, RTNE	SEC^{-1}	B4	1 halfword/ detector

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine / Subroutine	Units	Scale	Format
RRABC1	RRA1 (Byte 0), RRA2 (Byte 1), and RRC1 (byte 2) are the expected vehicle release rate through a downstream intersection for Links A1, B1, and C1, respectively	CONTR	*RTNY, RTND, RTNW	SEC/ VEH	B2 (EACH BYTE)	4 bytes/ controller
RTC	Real time clock	NA	*RTNY, RTNA, RTNB, RTNM	SEC	B5	1 word
R0, R1. . . . . , R15	General registers used for temporary storage	NA	ALL	ND	NA	1 word
SACC	Fifteen-minute accumulation of speed	LINK	*RTNY, RTNF, RTNH	SEC^{-1}	B4	1 halfword/link
SBAR	Smoothed link speeds	LINK	*RTNY, RTNF, RTNJ, RTNK, GOMTRY	MPH	B2	1 byte/link
SDAY	Accumulated average speed in a day	LINK	*RTNY, RTNH, RTNI	MPH	B3	1 halfword/link

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
SECCFC	Section-controller failure table. Number of failed controllers in each section	SECT	*RTNY, RTNH, RTNK, RTNM, KFMT, RTNQ	ND	B0	1 word/section
SECDFC	Section-detector failure table. Number of failed detectors in each section	SECT	*RTNY, RTNH, RTNK, RTNM, KFMT, RTNQ	ND	B0	1 word/section
SHTFLG	Shutdown flag	NA	*RTNY, RTNA, RTNH, RTNI, RTNL, RTNM, RTNW	ND	B0	1 word
SLFDPN	Starting location for desired pattern number. RAD starting point to read in pattern. INDEX = 4* (pattern number) + (section number)	INDEX	*RTNY, RTNW	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
SPLCHG	Split change flags	CONTR	*RTNY, RTNM, RTNR, RTNT	ND	B0	1 bit/controller**
SSFW	Single section for RTNW	NA	*RTNY, RTNM	ND	B0	1 word
ST	Temporary storage for link speed	NA	RTNF	MPH	B2	1 word
STARTA	Start of A-Phase green	CONTR	*RTNO, RTNR, RTNT	ND	B0	1 bit/controller**
STARTUPF	Flag set when 125 seconds have elapsed since RTNN was first entered	NA	RTNA1, RTNN, RTNY	ND	B0	1 word
STATE	State of CRT from RBM	NA	*RBM, CRTOUT	ND	B0	1 word
STBTIM	Standby time	CONTR	*RTNY, RTNR	SEC	B1	1 byte/controller
STBYD	Standby desired flag	CONTR	*RTNY, RTNM, RTNQ, RTNR	ND	B0	1 bit/controller**

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
STFLG2	Flags set at predicted green. Reset at stops threshold time for Zone 1	LINK	*RTNY, RTNC, RTND	ND	B0	1 bit/link
STFLG3	Flags set after predicted green when the total number of cars in Zones 0, 1, and 2 exceeds the capacity of Zones 0 and 1. Reset at stops threshold time for Zone 2.	LINK	*RTNY, RTNC, RTND	ND	B0	1 bit/link
STFLG4	Flags set at predicted red. Reset at predicted green	LINK	*RTNY, RTNC, RTND	ND	B0	1 bit/link
STFLG5	Flags used in BPS algorithm set after stops threshold time for Zone 1, if actual green has occurred	LINK	*RTNY, RTNC, RTND	ND	B0	1 bit/link
STOP1	Predicted stops threshold times for vehicles entering Zone 1	LINK	*RTNY, RTNC, RTND	ND	B1	1 byte/link
STOP2	Predicted stops threshold times for vehicles entering Zone 2.	LINK	*RTNY,	ND	B1	1 byte/link

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
STPTEMP	Temporary storage of stops	NA	*RTNY, RTNH	VEH	B0	1 word
SUMA	Temporary storage for	NA	RTNM, RTNV	SEC	B1	1 word
SUMABCT	SUMA (Byte 0), SUMB (Byte 1) and SUMC (Byte 2) are the summations of all fixed intervals which are part of A-Phase, B-Phase, and C-Phase green, respectively. Sumtotal (Byte 3) is a summation of all fixed intervals and the minimum duration of all adjustable intervals in a controller cycle which are not part of A-Phase, B-Phase, or C-Phase green	CONTR	*RTNY, RTNK, RTNM, RTNS, RTNW, UNPAK, SMSCYL, LPHASV	SEC	B1 (EACH BYTE)	4 bytes/ controller
SURDIS	Surveillance display code	NA	*RTNY, RTNM, RTNJ	ND	B0	1 word
SYSFLG	Flag set to 1 indicate array data update. Set to 0 to indicate single word update	NA	RTNM	ND	B0	1 word
TAMODE	Actual mode column of system status report	NA	RTNH	ND	B0	1 word/entry

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine / Subroutine	Units	Scale	Format
TCFAIL	Controller failures column of system status report	NA	RTNH	ND	B0	1 word/entry
TCODE	Type columen of failure status report	NA	RTNH	ND	B0	1 word/entry
TD	Pulse duration correction factor	NA	RTNC	SEC	B5	1 word
TDIFF	Temporary storage used in stop times' calculations	NA	RTNC	NA	NA	1 word
TDMODE	Desired mode column of system status report	NA	RTNH	ND	B0	1 word
TEMP	Absolute value of the difference between TVPLUSKO and MVPLUSKO	NA	RTNH	VPH	B0	1 word
TEMPSCF	Flags set for each cycle which has started before Routine E is entered	CYCLE CODE	RTNE	ND	NA	1 bit/cycle code
TEMPMOEF	Flag set when link being processed has an inoperative detector	LINK	RTNE	ND	NA	1 bit/detector
TEMPQ	Temporary storage used in stop times' calculations	NA	RTNC	NA	NA	1 word

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
TEMPS	Temporary storage of number of stops	NA	RTND	VEH	B1	1 word
TEMPZ	Temporary storage of zone counts	NA	RTND	VEH	B1	1 word
TFAIL	Detector failures column of system status report	NA	RTNH	ND	B0	1 word
THRESHOLD	Value assigned depending on control panel settings	NA	RTNJ, RTNM	ND	B0	1 word
THRESH	Flow data THRESHOLD	NA	*RTNY, RTNM, RTNJ	ND	B0	1 word
TIME	Time colume of failure status report	NA	RTNH	ND	B0	1 word
TIMEOUT	Timing cell	CONTR	*RTNY, RTNP	SEC	B1	1 byte/controller
TIMMCH	Time of mode change. Halfword '0' contains HRS Halfword '1' contains MIN	SECT	*RTNY, RTNH, RTNK, RTNM	HRS-MIN	NA	1 word/section
TK	Equals KCL		RTND, RTNM			

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
TKM	Minimum cycle length	CONTR	*RTNY, RTNS, RTNW, IRTN, UNPK1	SEC	B1	1 byte/controller
TLENGTH	Length of TSTART array	NA	*RTNY	ND	B0	1 word
TMDY	Time, month, day, year	NA	*RTNY	ND	EBC-DIC	1 word
TODC	Time of day count	NA	*RTNY	ND	B0	1 word
TOSU	First time RTNH was entered	HRS: MIN	*RTNI, RTNH	ND	EBC-DIC	1 word
TP	Temporary storage used in car count calculations	NA	RTNC	VEH	B3	1 word
TPAT	Pattern number column of system status report	NA	RTNH	ND	B0	1 word
TRANSF	Transition flags. Set in RTNW when a change in pattern is to be initiated. Reset in RTNT	CONTR	*RTNY, RTND, RTNK, RTNM, RTNP, RTNR, RTNT, RTNW, RTNX, PAKF	ND	B0	1 bit/controller**

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
TRTM	Travel time	NA	RTNH	SEC	B4	1 word
TSECT	Section number	NA	RTNH	ND	B0	1 word
TST	Temporary storage for GMINABC4 value	NA	RTNS	SEC	B1	1 word
TSTART	Pulse starting or ending times. Sign is negative after start of pulse and is positive after end of pulse	DET	*RTNY, RTNB, RTNE, RTNM	SEC	B5	1 word/detector
TT	Pulse duration (PRESENCE)	NA	*RTNY, RTNB, RTNC	SEC	B5	1 word
TTC	Corrected pulse duration	NA	RTNC	SEC	B5	1 word
TTIME	Time	NA	RTNH	ND	B0	1 word
TTSA	Time to stop array. Number of vehicles in a queue (QM)	PULSE LENGTH AND TTSASI	*RTNY, RTNC	VEH	B2	1 byte
TTSASI	Starting index for TTSA. List free flow speed Index (0, 1, 2)	LINK	*RTNY, RTNC	ND	B0	1 byte/link
TVPLUSKO	History volume plus weighted occupancy	NA	RTNH	VPH	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
VACC	Fifteen-minute accumulation of vehicle counts	LINK	*RTNY, RTNF, RTNH	VEH	B0	1 halfword/link
VDAT	Accumulated daily volume	LINK	*RTNY, RTNH, RTNI	VEH	B0	1 halfword/link
VHIST	History value of volume	LINK	*RTNY, RTNE, RTNF, RTNH, RTNW, RTNFT	VEH/ HR	B1	1 halfword/link
VO	Temporary storage for VOT value	NA	RTND, RTNV	VEH	B2	1 word
VOL	Smoothed vehicle volumes	LINK	*RTNY, RTND, RTNF, RTNH, RTNJ, RTNV, RTNW, LDMOE, GOMTRY	VEH/ HOUR	B1	1 halfword/link

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
VOT	Smoothed number of vehicles having exited from a link at downstream intersection during a controller cycle	LINK	*RTNY, RTND	VEH/ CYC	B2	1 byte/link
VPHT	Temporary storage of link volume	NA	RTNF	VEH/ HR	B1	1 word
VPLUSKO	History value of volume plus weighted occupancy subset of HSBUF array	NA	*RTNY, RTNM, RTNW	ND	NA	1 byte
W	Temporary storage used as detector change-of-state	NA	RTNB	ND	NA	1 word
WHSF	One-half second flag	NA	*RTNY, RTNA	ND	B0	1 word
WMAST	Priority of RTNW in mast	NA	*RTNY, RTNA, RTNH, RTNM, RTNW	ND	B0	1 word
WMINF	One-minute flag	NA	*RTNA, RTNL	ND	B0	1 word

*Defining routine.

Table 26. First Generation, UTCS, Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
WRKOFF	Working offset (Interval 1)	CONTR	*RTNY, RTNK, RTNM, RTNS, RTNT, RTNU, RTNW, RTNX, GETWORK, PKWRK	SEC	B1	1 byte/controller
WSF	One-second flag	NA	*RTNY, RTNA, RTND	ND	B0	1 word
YIELD	Label equated to IADVP	CONTR	RTNX	ND	B0	1 bit/controller**
ZD0	Car count in UTCS Zone 0	LINK	*RTNY, RTNC, RTND	VEH	B3	1 byte/link
ZD1	Car count in UTCS Zone 1	LINK	*RTNY, RTNC, RTND	VEH	B3	1 byte/link
ZD2	Car count in UTCS Zone 2	LINK	*RTNY, RTNC, RTND	VEH	B3	1 byte/link

*Defining routine.

**Word interleaved.

Table 26. First Generation, UTCS, Dynamic Arrays (Concluded)

Mnemonic	Definition	Index	Defining & Reference Routine/ Subroutine	Units	Scale	Format
ZF2	Estimate of queue extended beyond second detector for 2 or 3 detector links, where queue has not yet reached third detector	LINK	*RTNY, RTND	VEH	B1	1 byte/link
ZF3	Estimate of queue extended beyond third detector for 3 detector links, where stopped queue has reached third detector	LINK	*RTNY, RTND	VEH	B1	1 byte/link
ZLIM	Upper limit of zone index for section	NA	RTNH	ND	B0	1 word
ZOLD1	Zone 0 + Zone 1 counts up to stops threshold time	LINK	*RTNY, RTNC, RTND	VEH	B3	1 byte/link
ZOLD2	Zone 0 + Zone 1 + Zone 2 counts up to stops threshold time	LINK	*RTNY, RTNC, RTND	VEH	B3	1 byte/link
1MINCT	One-minute count	NA	RTNA	SEC	B5	1 word
15MINCYC	Time elapsed in current 15-minute period	NA	*RTNA, RTNK	SEC	B0	1 word
15TBUFF	Fifteen-minute MOE tape buffer for units and scaling see RTNH spec of Reference 4	1ST-INDEX	*RTNY, RTNH	ND	NA	1 halfword/entry

*Defining routine.

Table 27. Second Generation Dynamic Arrays

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
ABBCABCG	Working value of first generation dynamic array GABBCABC. Contents depend upon controller type as follows: Controller Type — Content K3PH = 0: Byte (0), time available for A- and B-Phases green Bytes (1) - (3), not used K3PH = 1: Byte (0), time available for A- and B-Phases green Byte (1), time available for B- and C-Phases green Byte (2), time available for A-, B-, and C-Phases green Byte (3), not used	CONTR	*RTNY, UNPK1, LOADAT, PATGENT, RTNM, RTNS	SEC	B1	4 bytes/ controller

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
	Controller Type / Content K3PH = 2: Byte (0), time available for A- and B-Phases green Byte (1), time available for B- and C-Phases green Bytes (2) and (3), not used K3PH = 3: Byte (0), time available for A- and B-Phases green Byte (1), time available for A- and C-Phases green Bytes (2) and (3), not used					
AVSPP	Average speed	NA	*RTNFT	MPH	B5	1 word
BIAS	Offset bias. The difference in time between start of true A-Phase green and start of Interval 1. A negative bias (A-Phase starts before Interval 1) is indicated by entry which exceeds 128	CONTR	*RTNY, PATGENT, LOADAT, PUTEOFF	SEC	B1	1 byte/ controller

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
CC2P	Accumulated detector car count used in second generation surveillance algorithm	DET	*RTNY, RTN5, RTNC	VEH	B0	1 byte/ detector
CICINH	MICRO loop control inhibit flags. Set to inhibit Routine LOCAL execution when new entry pattern data is being read into core from the RAD	CONTR	*RTNY, RTNV, RTNW	ND	B0	1 bit/ controller
COUNT	Accumulated free flow link count	NA	*RTN5, RTNFT	VEH	B0	1 word
CYC(I, J)	Cycle lengths to be considered in optimizing Subnetwork I		*MACRO, LSTSQS, RTSND	SEC	B0	8 x 5 words
DELT	Duration of leading movements for overlap controllers; i. e., time from start of A-Phase to start of C-Phase for K3PH = 2 type controllers; time from start B-Phase to start of C-Phase for K3PH = 3 type controllers. Negative numbers are indicated by entry which is greater than 128	CONTR	*RTNY, GOMTRY, UNPAK, PATGENT, LOADAT, RTNW	SEC	B1	1 byte/ controller

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
DELWN	Predicted delay for Subnetwork (I) if newly computed pattern is implemented		*MACRO, LSTSQS	ND	B0	1 word/ subnetwork
DELWO	Predicted delay for Subnetwork (I) if newly computed pattern is not implemented		*MACRO, LSTSQS	ND	B0	1 word/ subnetwork
DETM	Free flow detector fail flag	NA	*RTN5, RTNFT	ND	B0	1 word
ENTCYCLC	Background pattern cycle length code	CONTR	*RTNY, DELDAT, LOADAT, PATGENT, SMSCYL, MACA, PUTEOFF, ENTCYCL, RTNH, RTNX	ND	B0	1 byte/ controller
ENTCYSV	Cycle length code for current time-of-day pattern	CONTR	*RTNY, UNPK1, SMSCYL, RTNW	ND	B0	1 byte/ controller
GSBNET	Subnetwork number associated with each group	GROUP	*MACRO	ND	B0	1 word/group

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
IAA	Temporary storage for detector number	NA	*RTN5	ND	B0	1 word
IEA	Subnetwork number for controller	CONTR	*RTNY, LOADAT, RTNH, RTNW	ND	B0	1 byte/ controller
ILL	Temporary storage for link number	NA	*RTN5, RTNFT	ND	B0	1 word
INTF	Interface flags. Code is as follows: Code — Definition = 1 Groups 1 and 2 interface = 2 Groups 2 and 3 interface = 3 Groups 3 and 4 interface = 4 Groups 4 and 5 interface = 5 Groups 5 and 6 interface = 6 Section 2 and 1 interface = 7 Section 1 and 3 interface = 8 Section 3 and 4 interface = -1 No interface between subnetworks I and I + 1	SUBNET-WORK INTER-FACE	*MACRO, RTSND, INTFC	ND	B0	1 word/ subnetwork interface

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
	where: INTF(1) is interface between Subnetworks 1 and 2 INTF(2) is interface between Subnetworks 2 and 3 INTF(3) is interface between Subnetworks 3 and 4 INTF(4) is interface between Subnetworks 4 and 5 INTF(5) is interface between Subnetworks 5 and 6					
INTFO	Value of INTF(I) from previous pass through MACRO loop	SUBNET-WORK INTER-FACE	*MACRO, RTSND	ND	B0	1 word/ subnetwork interface

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine / Subroutine	Units	Scale	Format
IO	Entry offsets and controller phase order code. Byte 0, true A-Phase offset (not necessarily Interval 1 offset) Byte 1, B-Phase offset Byte 2, C-Phase offset Byte 3, phase order code for K3PH = 0 and K3PH = 1 controllers Controller Type / Code (Byte 3) K3PH = 0: 0 indicates A, B 1 indicates B, A K3PH = 1: 0 indicates A, B, C 1 indicates A, C, B	CONTR	*RTNY, RTNS, RTNT, RTNV, GOMTRY, LOADAT, PATGENT, DELDAT, PUTEOFF, MACA, GETEO, RTNW, RTNX, RTNH	SEC	B1	4 bytes / controller
IRADRD	Flag set in RTNW when new pattern is read from RAD. Reset in MACRO	NA	*RTNY, RTNW, MACRO	ND	B0	1 word

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
IREOPT	Flag set when MACRO loop is to compute and implement new signal timing pattern. Reoptimization occurs if at least one section is in the Traffic Responsive Mode and at least one of the following is true: a) A new Time-of-Day pattern has been read into core from RAD b) The operator has input a mode change request for any section since the last pass through MACRO c) A new subnetwork structure has been configured d) The optimal cycle length computed for any subnetwork is different from that of the last pass e) The reduction is predicted change in delay between using the new pattern versus the current pattern is greater than CHNGTH	NA	*MACRO, RTSND, LSTSQS	ND	B0	1 word

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
ISB(I, J)	Groups included in Subnetwork J	NA	*MACRO, RTSND, INTFC, LSTSQS, TRNPAR	ND	B0	8 x 8 words
ISO	Subnetwork number in which group was operating in on last pass through MACRO loop	NA	*MACRO, LSTSQS	ND	B0	1 word/group
ITRDLY	Traffic responsive delay flag. Set after traffic responsive pattern has been computed and implemented to indicate that reoptimization should not occur on next pass through the MACRO loop	NA	*MACRO, LSTSQS, RTNM	ND	B0	1 word
LEVL5A	Flag set in RTNA when Priority Level 5 routines are in execution	NA	*RTNY, RTNA, RTNM, RTNW	ND	B0	1 word
LEVL5O	Value of LEV5SF from previous pass through MACRO loop	SECT	*MACRO	ND	B0	1 word/ section
LEV5SF	Flags set at exit of RTNW to indicate to MACRO loop that operator has selected a section for Traffic Responsive Mode of control	SEC	*RTNY, RTNW	ND	B0	1 word/ section

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
LSQ	Flags set in RTNW when operator has selected Traffic Responsive Mode for a section	SECT	*RTNY, RTNW	ND	B0	1 byte/ section
NCYCLE	Number of cycle lengths to be considered in optimizing Subnetwork I	SUBNET-WORK	*MACRO, RTSND, LSTSQS	ND	B0	1 word/ subnetwork
NDD	Number of detectors processed on link	NA	*RTN5	ND	B0	1 word
NII	Storage for number of detectors on link	NA	*RTN5, RTNFT	ND	B0	1 word
NSUBO	Number of subnetworks formed on last pass through MACRO loop	NA	*MACRO, RTSND	ND	B0	1 word
OCCPP	Free-flow link occupancy to be stored in tape buffer	NA	*RTN5, RTNFT	PER-CENT	B0	1 word
OPCP	Average link occupancy to be stored in tape buffer	NA	*RTNFT	PER-CENT	B0	1 word
OPTIN	Number of optimization intervals in an hour	NA	*RTNY, RTNFT	ND	B0	1 word

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
PS	Predicted speed on each first generation link	LINK	*MACRO, PREDICT, LSTSQS	MPH	B0	1 word/link
PV	Predicted volume on each first generation link	LINK	*MACRO, PREDICT, RTSND, LSTSQS	VPH	B0	1 word/link
RAWOCP	Accumulated raw occupancy for each detector	DET	*RTNY RTNS, RTNC	SEC	B5	1 halfword/ detector
RAWSPP	Accumulated raw speed for each detector	DET	*RTNY, RTNS, RTNC	/SEC	B4	1 halfword/ detector
RLNKC	Raw link count	NA	*RTN5, RTNFT	VEH	B0	1 word
RLNKO	Raw link occupancy	NA	*RTN5, RTNFT	SEC	B1	1 word
RLSPEED	Raw link speed	NA	*RTN5, RTNFT	/SEC	B4	1 word

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
SABC	Summation of fixed intervals in each phase of controller cycle Byte 0, sum of fixed intervals in A-Phase Byte 1, sum of fixed intervals in B-Phase Byte 2, sum of fixed intervals in C-Phase Byte 3, sum of fixed intervals in controller cycle which are not part of A-, B-, or C-Phases	CONTR	*RTNY, UNPAK, RTNV, LOADAT, PATGENT, GTSMT, RTNM	SECS	B1	1 word/ controller
SACCP	Average link speed	NA	*RTN5	MPH	B0	1 word
SAVRO	Storage for address that FORTRAN routines use to access Data Control Blocks	NA	*RTNY, RTNA, RTNV	ND	B0	1 word

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
SMTABC	Time duration of fixed intervals (clearance intervals which are not part of a phase) at the end of each phase Byte 0, duration of clearance interval at end of A-Phase Byte 1, duration of clearance interval at end of B-Phase Byte 2, duration of clearance interval at end of C-Phase Byte 3, sum of Bytes 0 through 3	CONTR	*RTNY, RTNV, UNPAK, LOADAT, PATGENT, SMSCYL, LPHASV	SECS	B1	1 word/ controller
SPEED	Free flow link speed	NA	*RTN5, RTNFT	MPH	B0	1 word
SPLIT	Time-of-day split (entry) for each controller: Halfword 0, A-Phase split Halfword 1, C-Phase split (K3PH = 1 type controllers only)	CONTR	*RTNY, GSPLEL PATGENT, SPLTWP, MACA	PER-CENT x 10	B0	1 word/ controller

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
SPLITW	Working split for each controller: Halfword 0, A-Phase split Halfword 1, C-Phase split (K3PH = 1 type controllers only)	CONTR	*RTNY, GSPLWL, SPLPKL, LOADAT, PATGENT, SPLTWP, SPLPK, GETSPLT	PER-CENT x 10	B0	1 word/ controller
SUMACBC	Sum of fixed intervals in overlapped portion of controller cycle. Not used for K3PH = 0, or K3PH = 1 type controllers	CONTR	*RTNY, RTNV, UNPAK, LOADAT, PATGENT, SMSCYL, GSACBC	SECS	B1	1 byte/ controller
TBUFF	Five minute tape buffer. See RTNFT documentation for contents	LINK	*RTN5, LDMOE5 RTNFT, PREDICT	SEE DESCR	B0	5 words/link
TINDEX	Tape buffer index	NA	*RTN5, RTNFT	ND	B0	1 word
VACCP	Average volume on link	NA	*RTNFT	VPH	B0	1 word

*Allocating routine/subroutine.

Table 27. Second Generation Dynamic Arrays (Continued)

Mnemonic	Definition	Index	Allocating & Reference Routine / Subroutine	Units	Scale	Format
WRKOFS	Working offsets for each controller: Byte 0, true A-Phase offset (not necessarily Interval 1 offset) Byte 1, B-Phase offset Byte 2, C-Phase offset Byte 3, not used	CONTR	*RTNY, RTNT, RTNS, RTNV, WORK, PKWRK, GOMTRY, RTNH, RTNX	SECS	B1	4 bytes / controller
5END	Flag set in RTNQ to indicate to the executive RTNA that MACRO loop is to be scheduled to write predictor shutdown report	NA	*RTNY, RTNA, RTNQ	ND	B0	1 word
5END1	Flag set in RTNA to indicate to MACRO loop to write the predictor shutdown report	NA	*RTNY, SHTDWN, RTNA	ND	B0	1 word
5END2	Flag set in RTNI to indicate to MACRO loop to write the end-of-day predictor report	NA	*RTNY, SHTDWN, RTNI	ND	B0	1 word
5MINCT	Counter in RTNA used to synchronize the scheduling of the Priority Level 5 MACRO loop	NA	*RTNY, RTNA	SECS	B0	1 word

*Allocating routine / subroutine.

Table 27. Second Generation Dynamic Arrays (Concluded)

Mnemonic	Definition	Index	Allocating & Reference Routine/ Subroutine	Units	Scale	Format
5MINF	Flag set in RTNA to indicate to the RTNA one-half second logic that hardware interrupt should be triggered to bring Priority Level 5 tasks into execution. Flag is set when 5MINCT times out	NA	*RTNY, RTNA	ND	B0	1 word
10VWD9	Used to convert OCCP to percent. Value is function of MACRO loop execution frequency	NA	*RTNFT	ND	B0	1 word

*Allocating routine/subroutine.

Table 28. COMMON Blocks in UTCS/TANSTP Software

Block Name	Arrays in Block	Routines Which Access Block	Function of Block
BLK	ISB (8, 8) NCYCLE (8) CYC (8, 5)	MACRO, RTNSD, LSTSQS	Subnetwork groupings and subnetwork candidate cycle lengths
BLKIF	INTFLG	MACRO	Not used
BLKIFC	INTF (5)	MACRO, RTSND, INTFC, TRNPAR	Flags which indicate groups or sections which comprise each subnetwork interface
BLKIF1	GSECT (6) GRPIFC (10) NGRIPF GSBNET (6) NINTFC NGRPSS (8) INFCR1 (5)	MACRO, RTSND, INTFC, TRNPAR	Detailed subnetwork configuration data
BLKLV5	LEVL5A ITRDLY	MACRO, RTNA	Priority Level 1/Priority Level 5 communication flags
BLK1	LEV5SF (4) IRADRD	MACRO, RTNW	Traffic Responsive Mode communication from Priority Level 3 to Priority Level 5
BLK2	INTRS (45)	MACRO, LSTSQS, TRNPAR	Temporary storage for controllers contained in each group
BLK3	PV (262) PS (263)	MACRO, PREDICT, LSTSQS, RTSND, INTFC	Predictor volume and speed outputs

Table 28. COMMON Blocks in UTCS/TANSTP Software (Continued)

Block Name	Arrays in Block	Routines Which Access Block	Function of Block
BLK4	DATA (144, 8) NL	MACRO, PREDICT	Predictor Communication Block
BLK5	IPRD CYCMAX CYCMIN OCTH OCCW SECDFF CHNGTH SPLCHT TOL NMAX NGRPS OPTIME SATHRH SATHLW	MACRO, RTSND, LSTSQS INTFC	Data array giving user capability to specify limits on MACRO loop control
BLK6	IWORK (15, 6)	MACRO, RTSND	Subnetwork Determination Debug arrays
BLK7	LEV5O (4) CYCLO (8) INTFO (5) NSUBO IREOPT	MACRO, RTSND, LSTSQS	Communication between consecutive passes through MACRO loop for reoptimization decision logic
BLK20	SECIFC (6)	MACRO, RTSND, INTFC	Array giving Possible Section Interfaces

Table 28. COMMON Blocks in UTCS/TANSTP Software (Concluded)

Block Name	Arrays in Block	Routines Which Access Block	Function of Block
BLK46	ISO (8)	MACRO, LSTSQS	Communication Block between successive passes of Routine LSTSQS. Used in Reoptimization Decision Logic.
BLK50	DELWO (6) DELWN (6)	MACRO, RTSND	Communication Block between successive passes of Routine LSTSQS for Decision Logic on Implementing new pattern
BLK89	ITSAVE HOLDF CP (275) CS (275)	MACRO, PREDICT	Communication Block between successive passes of Routine PREDICT
BLK99	LWORK (100)	LOCAL	Local intersection control debug array

6.2 BASIC ELEMENTS OF UTCS/TANSTP DATA BASE AND GRAPHICAL REPRESENTATION OF UTCS/TANSTP DATA BASE FOR WASHINGTON, D. C., UTCS NETWORK

Basic elements of the UTCS/TANSTP data base structure are as follows:

a) Controllers

b) Detectors

c) First generation, UTCS, links

d) Second generation, TANSTP, links

e) Groups

f) Sections

g) Group interfaces

h) Section interfaces

The definition of each of the above elements and the data base arrays which define them are given below.

6.2.1 Controllers

Every traffic signal controller requires a number to be assigned to it in the software. This assignment is made in the array, SECCONT, which is a list of the controller numbers assigned in the system. The controller numbers are grouped according to section in this array.

6.2.2 Detectors

Every vehicle detector in the system also requires a number to be assigned to it. This assignment is made implicitily in the array, DWL. Each entry in this array corresponds to a detector. Entries are sequential with respect to dector number. That is, the first entry corresponds to Detector 1, the second to Detecor 2, and so on. The entries indicate the first generation link to which the detector is assigned. The position of the detector relative to the stop bar is given by the array DWLP.

6.2.3 First Generation Links

Every traffic lane in the system which has vehicle detectors on it is assigned a first generation, UTCS, link number in the software. In the first generation link numbering scheme, only those lanes having detectors have link numbers assigned to them. This assignment is made in the

array, SECTLKN, which is a list of link numbers grouped by section. These links also have controller assignments defined by the combination of arrays CONTLN and LSI. Also the array, LNKCON, gives the controller number to which each link is assigned.

6.2.4 Second Generation Links

In order for the TANSTP optimization routine, LSTSQS, to perform the mathematical operations to arrive at an optimal signal timing plan, every approach to a controller to be optimized by the software requires a link number. A link number is assigned to each traffic flow direction for each approach. Thus one-way streets will have only one link number for each approach. Two-way streets will have two link number assignments, one for each flow direction.

The second generation link number assignments are made implicitly in the software via entries in the array LMNO. This array gives the controller numbers upstream and downstream of each second generation link. Entries in the array are sequential with respect to link number. That is, the first entry corresponds to Link 1, the second to Link 2, and so on.

The traffic flow characteristics of speed and volume on each second generation link must also be input to the optimization algorithm. The traffic flow characteristics are acquired from the vehicle detectors in the system. Thus a first generation link number must be assigned to each second generation link. This assignment is made through the array 1GLNK.

6.2.5 Groups

Controllers are assigned to groups in the second generation, TANSTP, software. A group of controllers will always operate on a common cycle length while under second generation, Traffic Responsive Control. The controllers assigned to each group are given by the arrays GCONT, GCONTSI, and NCIGRP.

Each group is also assigned to a section. This assignment is made in the array GSECT.

If adjacent groups of controllers are both in the Traffic Responsive Mode of control, they may combine to form a subnetwork. Each subnetwork will operate on a common cycle length as described in Section 3.4.

6.2.6 Sections

A section is defined as a set of controllers and their associated links which will always operate in the same mode of control. The possible modes of control are Standby, Manual, Time-of-Day, or Traffic Responsive. Note that all controllers in a section are not required to operate on a common cycle length.

The controllers assigned to each section are given by the arrays SECCONT, SECCONSI, and NC. The links assigned to each section are given by the arrays SECTLKN and LINKSI.

6.2.7 Subnetwork Interfaces

In the second generation, Traffic Responsive Mode of control, groups of controllers will combine to form subnetworks. Subsequently, an attempt is made to minimize vehicle delay across the subnetwork boundaries as described in Section 3.6. These boundaries are called subnetwork interfaces. The boundaries are composed of the second generation links traversing the boundary and the controllers upstream and downstream of these links.

The delay minimization procedure classifies one subnetwork on one side of the interface as a base subnetwork. The subnetwork on the other side of the interface is classified as the slaved subnetwork. That is, the offsets generated by Routine LSTSQS for all controllers in the slaved subnetwork are adjusted by a constant such that a reasonable progression will be maintained on the links connecting the two subnetworks.

Subnetwork interface boundaries will coincide with particular group interface boundaries in all instances. However, two or more groups in the slaved subnetwork may interface with one or more groups in the base subnetwork. In these instances, the subnetwork interface is composed of more than one group interface. To handle cases of this sort, section interfaces are defined in the second generation software which are composed of two or more group interfaces. The section interfaces have priority over the group interfaces if the subnetwork interface coincides with the boundary between sections. The decision of whether to minimize delay across group interfaces or section interfaces is made through testing the value of the interface flags, INTF. All possible values of these flags and their meanings are given in Table 15.

6.2.7.1 Group Interfaces

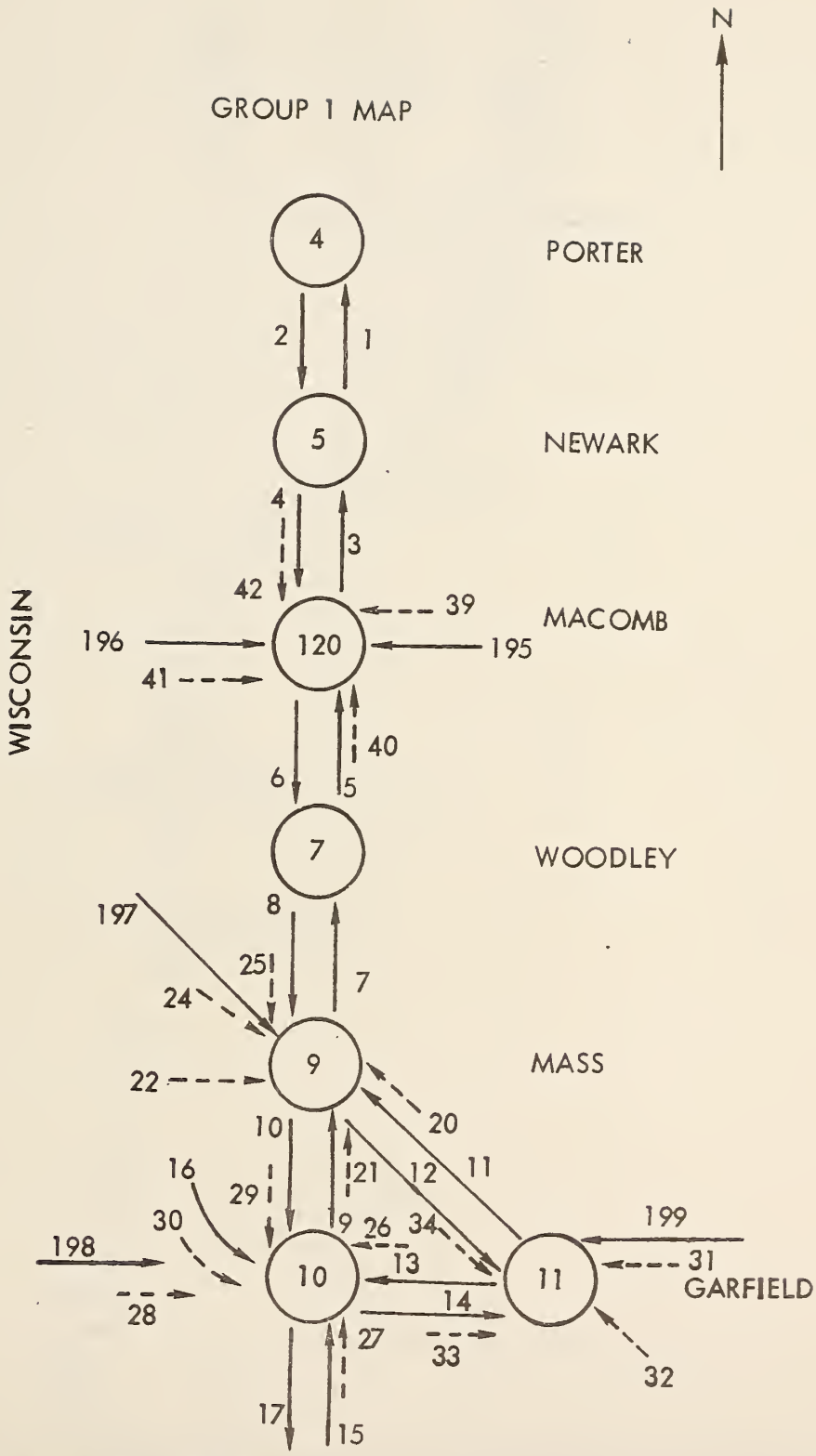
The number of group interfaces and the groups on both sides of each group interface are given by the arrays NGRIPF and GRPIFC, respectively. These arrays are located in Routine MACRO. The section interface corresponding to each group interface is given by the array INFCR1 also located in Routine MACRO. The controllers and links associated with each group interface are defined in Routine INTFC by the arrays INTN, INTN2, LINKN, NGRPI, and NINTF.

6.2.7.2 Section Interfaces

The sections on both sides of each section interface are given the array SECIFC in Routine MACRO. The controllers and links with each section interface are defined in Routine INTFC by the arrays INTNS, INTN2S, LINKNS, NGRPIS, and NINTFS.

6.2.8 Graphical Representation of UTCS/TANSTP Data Base for Washington, D. C., UTCS Network

The UTCS network links and nodes optimized by the second generation software are shown in Figure 75 through 80. As depicted by these figures, the network is broken down into six groups of controllers. Section 1 contains Group 3, Section 2 contains Groups 1 and 2, Section 3 contains Groups 4 and 5, and Section 4 contains Group 6. The first generation controller numbers contained in each group are shown on the figures. Also, shown are the second generation link numbers (solid lines) and the first generation link numbers (dashed lines) associated with each controller. Second generation link numbers are used by Subroutine LLGTHF to access values of link lengths, link phasing, and link importance factors. First generation link numbers are used by Subroutine LDMOE5 to access surveillance data. Routine RTSND (Real-Time Subnetwork Determination) obtains traffic surveillance data from detectors at the instrumented controllers in each core group. Based on this data, one or more groups may be combined into one subnetwork to be optimized by the second generation optimization Routine LSTSQS. The technique used for determining subnetwork configurations is described in Section 3.4 of this document. It should be reiterated, that only those groups which are contained in sections which are selected for the Traffic Responsive Mode of control are considered by Routine RTSND in determining subnetworks.



---Dotted line indicates first generation link

Figure 75. Controllers and Links in UTCS Network Group 1

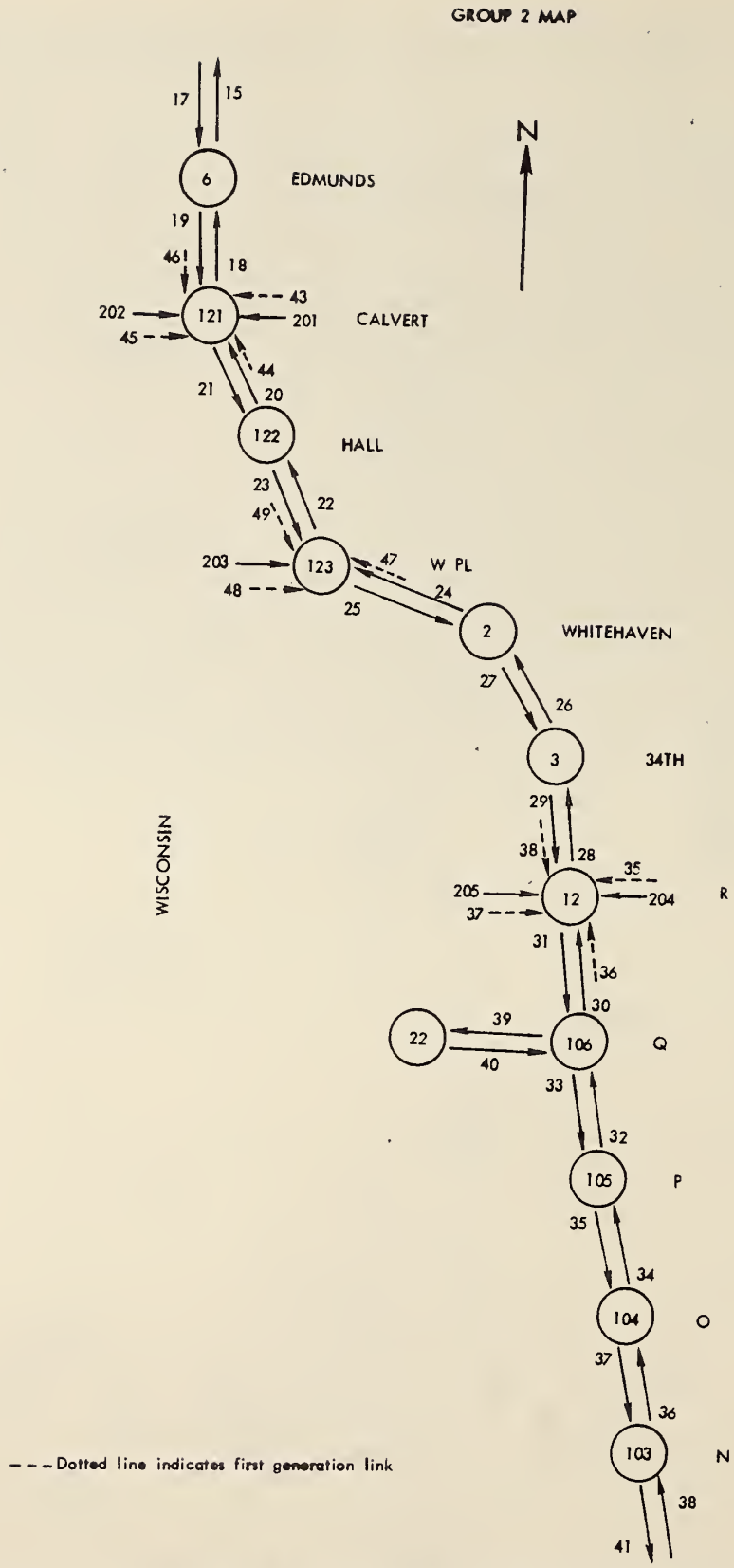
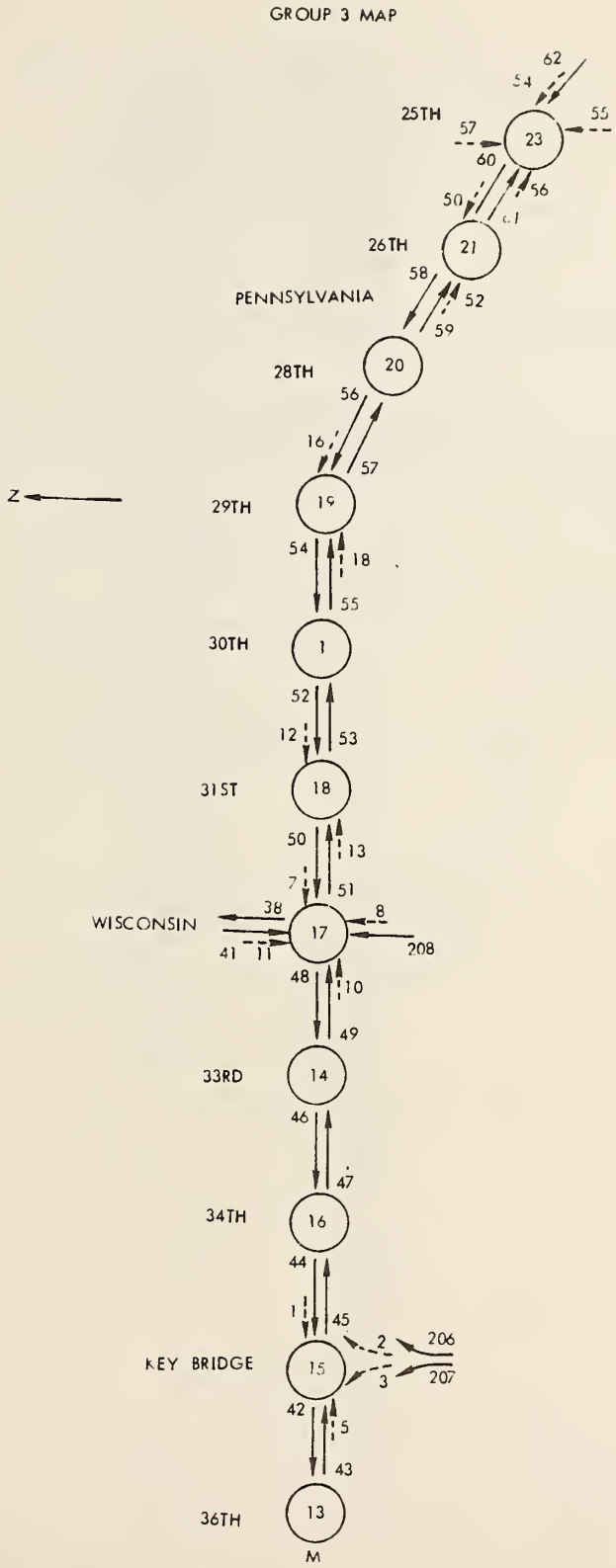


Figure 76. Controllers and Links in UTCS Network Group 2



--- Dotted line indicates first generation link

Figure 77. Controllers and Links in UTCS Network Group 3

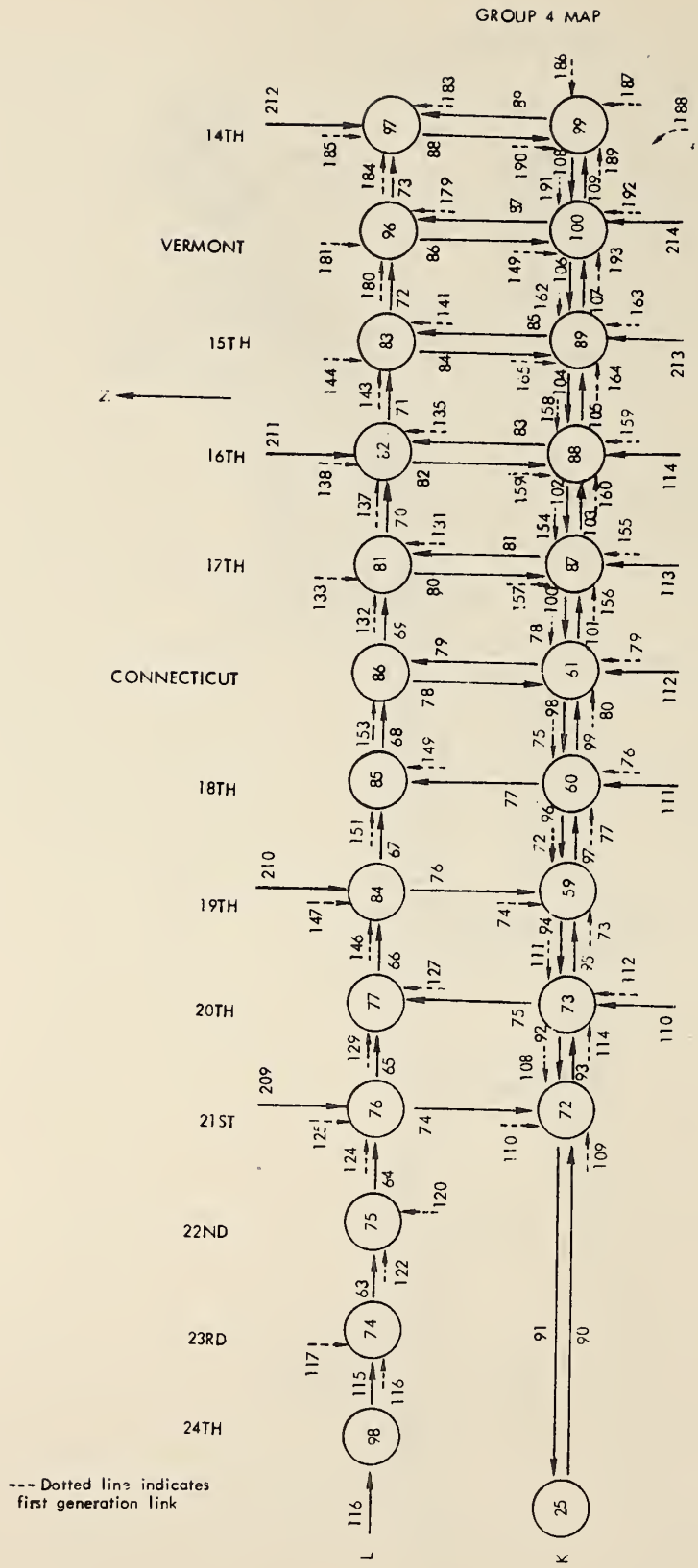


Figure 78. Controllers and Links in UTCS Network Group 4

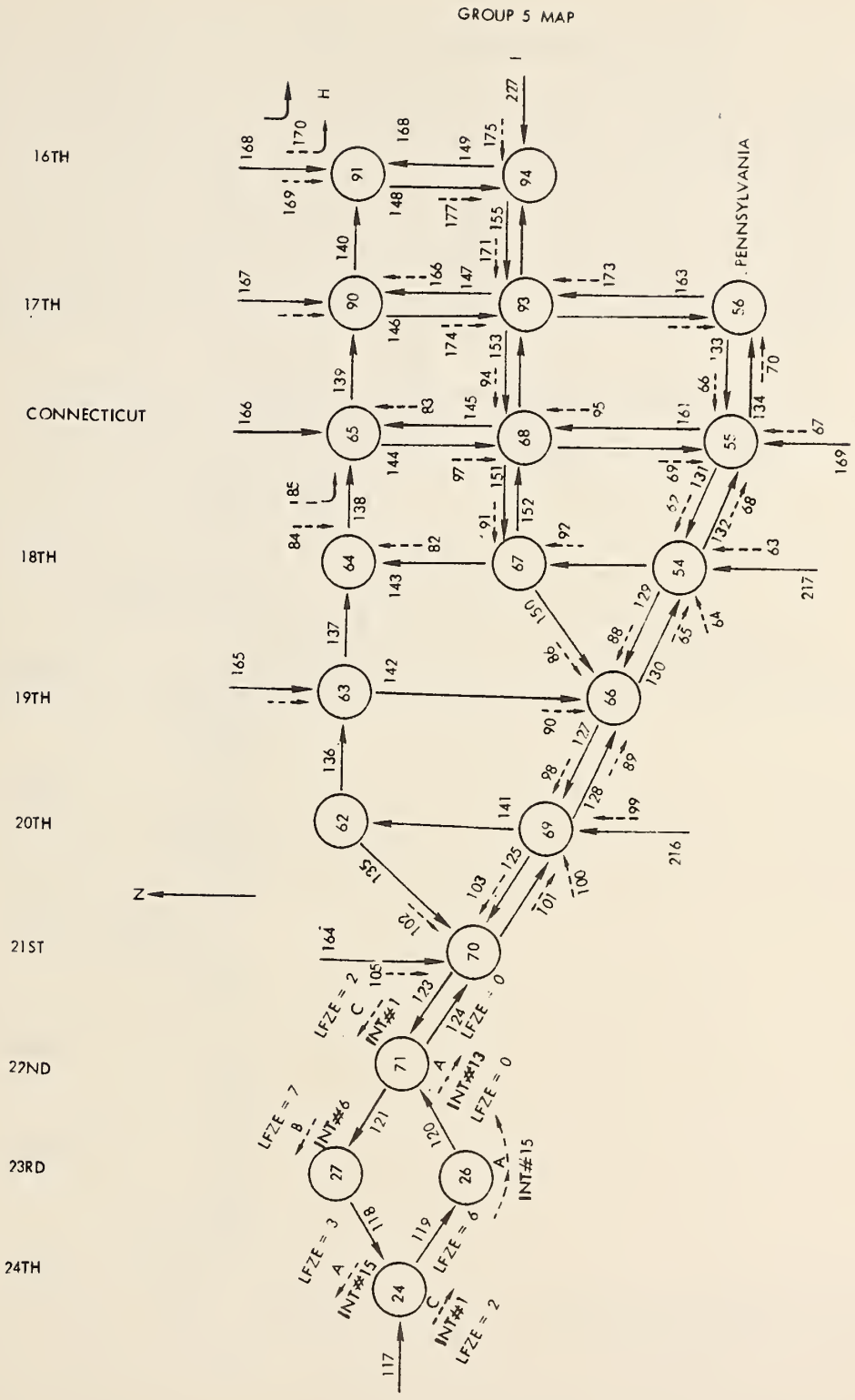


Figure 79. Controllers and Links in UTCS Network Group 5

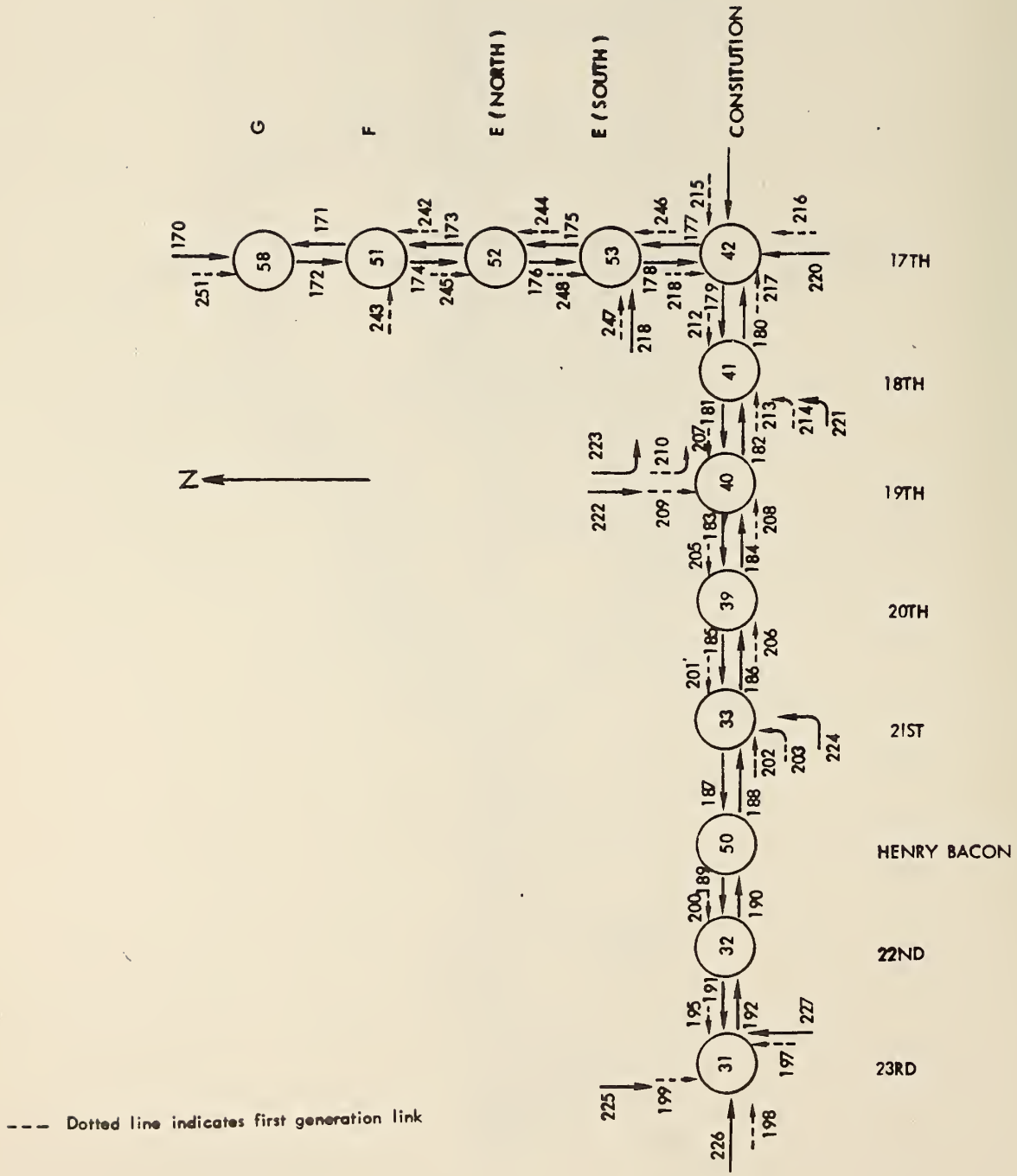


Figure 80. Controllers and Links in UTCS Network Group 6

6.3 UTCS/TANSTP DATA BASE UPDATE PROCEDURES

This section provides examples of data base changes normally expected to be encountered by the traffic engineer to accommodate changes in the traffic signal network to be controlled by the UTCS/TANSTP software system. These examples are not intended to be all inclusive of every change which may be encountered. For example, changes required of the software to drive a wall map which has been modified are not covered. It is recommended that changes of this type be left to the skilled programmer/analyst because of their complexity. The specific examples given here cover the following common changes in the traffic signal network:

1) Addition of a first generation link.

2) Addition of a second generation link.

3) Addition of a controller.

4) Change in group assignment for a controller.

5) Change in section assignment for a controller.

All data base arrays referred to in the following subsections are defined in Tables 23 through 27.

6.3.1 Procedures for Adding a First Generation Link

The procedures for inserting an additional first generation link are as follows:

1) Determine section in which the link is to be added.

2) Assign a number to the link. Be careful that this number is not already assigned to another link.

3) Increase the number of links in the system, NUMLNK, by unity in Routine RTNY.

4) Insert the link number in the section link array, SECTLKN, of Routine RTNY. Remember that the link numbers entered in this array are grouped by section. Therefore care must be taken to insure that the new link is inserted in the section to which the new link is assigned. Also remember that the entries in this array are zero based. That is, Link 1 is entered as a 0, Link 2 as 1, and so on.

5) Expand the number of links in the section to which the new link belongs through adjusting the section starting link index array, LINKSI, in Routine RTNY. Remember that this array is zero based.

6) Use the same procedure as 4) in updating the SECLKN array in Subroutine PTNRAD of the off-line pattern program, PATGEN. (See Section 7.1 for description of PATGEN.) The array SECLKN is one based.

7) Use the same procedure as 5) in updating the arrays LNKST and LNKEND in Subroutine PTNRAD of Program PATGEN. These arrays are the link starting index and the link ending index, respectively, for each section. These arrays are one based.

8) Update the IBYTE array of Subroutine PTNRAD of the PATGEN program. This array has an entry for each section of controllers in the system. The first entry is for Section 1, the second for Section 2, and so on. Each entry gives the length of the pattern data for the section in terms of number of computer words. The entry corresponding to the section to which the new link is associated should be increased by unity. The reader should consult Section 7.1 for more detailed information on updating the data arrays of Subroutine PTNRAD.

9) Update the appropriate entry in the LFDPN array of Routine RTNY to be consistent with the IBYTE array. Like the IBYTE array, each entry in the LFDPN array gives the length of the pattern data for each section of controllers. Unlike the IBYTE array, however, LFDPN gives this length in number of bytes. (The number of bytes is four times the number of words.)

10) Update the PATGEN data deck to include historical volume and occupancy for the new link for each manual and time-of-day pattern.

11) Determine the controller number to which the new link is associated and update the controller link array, CONTLN, and the link controller array, LNKCON, in Routine RTNY. These are zero based arrays. If the link is not a critical link, i. e., it has only one detector, it should be the last entry in the CONTLN array for the controller to which it is associated. The procedure for critical links will be described later.

12) Update the link starting index array, LSI, in Routine RTNY, for all controllers having instrumented approaches and having numbers above the controller number to which the new link is associated. Each entry in LSI for these controllers should be increased by unity.

13) Insert entries into the following arrays of Routine RTNY which describe the characteristics of the new link:

    LINKDI - Link direction.

    LNKPT - Phase of controller to which link is associated.

14) Assign a detector number to each new detector associated with each link. Insure that these detector numbers are not already assigned. Insert the new link number into the detector link array, DWL, of Routine RTNY for each new detector. The array DWL is sequential with respect to detector number. That is, the first entry into DWL corresponds to Detector 1, the second corresponds to Detector 2, and so on. The link numbers entered into DWL are zero based.

15) Based upon the number of new detectors assigned to the new link, update the NDALNK and NDOL arrays accordingly. These arrays are contained in Routine RTNY and entries are sequential with respect to link number.

16) Place appropriate entries into the DWLP array of Routine RTNY for each new detector. This array is sequential with respect to detector number.

17) Increase the value of the parameter NUMVED in Routine RTNY by the number of detectors added.

18) If the new link is not a critical link, i. e., it has less than two detectors on it, the necessary updates are now complete. The procedures of Sections 8. 2. 2 and 8. 2. 3 should be used to assemble and the software system. (The PATGEN source deck requires assembling and a new PATGEN binary save tape is required to be generated.) If the new link is a critical link, i. e., it has more than one detector, the following additional steps are required.

19) If the new link is a critical link and the controller to which it is associated has no critical links in the current network configuration, the following arrays in Subroutine PTNRAD require updating in addition to the arrays IBYTE, LNKST, LNKEND, and SECLKN:

ICPOS - Starting byte position in section pattern array for each controller.

HISTLOC - Starting byte position in section pattern array for historical volume and occupancy data corresponding to lowest numbered link in section.

The reader should refer to Section 7. 1 for details on making the updates to ICPOS, HISTLOC, and IBYTE.

20) Update the LFDPN array of Routine RTNY to be consistent with IBYTE.

21) The array giving the number of critical links for each controller, NQL, requires updating. The entry in this array corresponding to the controller to which the new link is associated requires that it be incremented by one. This array is located in Routine RTNY.

22) If the new link is a critical link, disregard the procedure concerning the array CONTLN in Step 11) above and use the following rules to update array CONTLN:

a) The entries in the CONTLN array for a given controller should first list all critical links associated with the controller. The critical links are then followed by the noncritical links associated with the controller if such links exist.

b) The order of the critical link entries into the array CONTLN for a given controller depend upon the controller phase to which each link is associated. A-Phase links are listed first, B-Phase links are listed next, and finally the C-Phase links are listed. No more than two critical links may be associated with A- and B-Phases and no more than one critical link may be associated with C-Phase.

23) The entries in the array, LDSET, corresponding to the controller to which the new link is associated require updating using the rules listed below. The array LDSET contains four bytes (one word) for each controller. The data entries in each byte depend upon the number of critical links associated with each controller phase as follows:

Number of Critical Links per Phase			LDSET Entries for Controller			
A-Phase	B-Phase	C-Phase	Byte 0	Byte 1	Byte 2	Byte 3
1	1	0	0	1	0	0
1	1	1	0	1	0	2
2	1	0	1	2	0	0
2	1	1	1	2	0	3
1	2	0	0	1	2	0
1	2	1	0	1	2	3
2	2	0	1	2	3	0
2	2	1	1	2	3	4

From the above table, it is apparent that each byte entry in LDSET corresponds to a controller phase as follows:

Byte	Controller Phase
0	A_2
1	B_1
2	B_2
3	C_1

A value of zero entered into any byte of LDSET indicates that no critical link is associated with that particular controller phase. A nonzero entry indicates how much LSI should be incremented in order to retrieve the critical link number associated with that particular phase from the CONTLN array. LSI is a zero based array and points to the link entry in the CONTLN array associated with Phase A_1.

The LDSET array is contained in Routine RTNY.

24) If the new link has a corresponding second generation link, the entry in the 1GLNK array corresponding to the second generation link number should be changed to reflect the new first generation link. The 1GLNK array is one based and is contained in Subroutine LLGTHF.

25) The necessary updates are now complete. The procedures of Sections 8.2.1, 8.2.2, and 8.2.3 should be used to assemble and link the software system.

6.3.2 Procedures for Adding a Second Generation Link

The procedures for adding a second generation link are as follows:

1) Determine the group number to which the new link is associated.

2) Make an entry into the array, LMNO, for this link. This array is contained in Subroutine LINKD and is ordered by the group number to which each link is associated. That is, entries for links associated with Group 1 appear first, Group 2 links next, and so on. Note that inserting data for a new link in array LMNO implicitly renumbers all subsequent second generation links. However, the software in Routine LSTSQS automatically accounts for this implicit renumbering and the analyst need not be concerned about it.

   The array LMNO contains four bytes of data for each link. Byte 0 contains the second upstream controller number, Byte 1 contains the first upstream controller number, Byte 2 contains the first downstream controller number, Byte 3 contains the second downstream controller number. The reader should consult Figure 10 of Section 3.5 for the meaning of first and second controller numbers.

3) Entries for the new link should be inserted into the following arrays:

   1GLNK, LENGTH, IALPH, JALPH, and LFZE

   Like LMNO, these arrays are ordered by the group number to which each link is associated. Care should be exercised to enter the new data into each of these arrays at locations corresponding to the new entry into the LMNO array.

4) The arrays N2GCL and 2GCLSI need to be updated to reflect the insertion of new second generation link.

5) A precautionary check should be made to insure that the current dimensions of the link oriented data are not exceeded in Routine LSTSQS. The current dimensions are set to 175 and must accommodate all real and continuity links for the largest subnetwork which can be formed. The reader should refer to Section 3.5 for a discussion of continuity links. If the current dimensions are exceeded, they must be changed to accommodate all network links.

6) The necessary updates for adding a second generation link are now completed. The procedures of Section 8. 2. 1 should be used to reassemble and link the software system.

Before closing this subsection one final point is in order. This point is that space for future links can be saved in the data base by inserting 'dummy' links in the arrays LMNO, 1GLNK, LENGTH, IALPH, JALPH, and LFZE. A dummy link is denoted by entering zero into each byte of the LMNO array corresponding to the dummy link. The dummy links will be deleted from the network by Routine LSTSQS. However, in order for this deletion logic to work properly, a nonzero entry must appear in the elements of the 1GLNK array corresponding to each dummy link. If this procedure is not followed, the Traffic Responsive traffic signal timing plans will not be generated correctly.

6.3.3 Procedures for Adding a Controller

The procedures for inserting an additional controller are as follows:

1) Assign a number to the controller. Be sure that this number has not already been assigned.

2) Determine the section to which the new controller belongs.

3) Update the CTLSEC and CTLSYS arrays of Routine RTNY to include the new controller. These arrays have one bit for each controller. The bit corresponding to the new controller should be set to 1.

4) Update the proper element of the array NC of Routine RTNY. This array lists the number of controllers in each section.

5) Update the number of controllers in the system by incrementing the parameter NUMCTL of Routine RTNY. (Note: This may not be necessary if one of the available dummy controller entries reserved in the current software is used for the new controller. If a dummy controller slot is used for the new controller, the controller number must be deleted from the nonexistent controller array NOCONT, and the number of nonexistent controllers, NONCON$, must be decremented by one. The entry in NUMCTL includes the number of actual and dummy controllers. NOCONT and NONCON$ are contained in Routine RTNK.)

6) Insert the new controller into the section controller array, SECCONT. The controllers are grouped by the section to which they are assigned in this array. Also update the section controller starting index array, SECCONSI, to reflect the fact that the number of controllers in one section has changed and therefore the starting index for the sections subsequent to this section must be increased by one. The SECCONT and SECCONSI arrays are contained in Routine RTNY and are zero based.

7) If the new controller is semiactuated, the bit corresponding to this controller must be set to zero in the flagword SEMMSK of Routine RTNY.

8) Update the arrays of Routine RTNY describing the controller characteristics. These are as follows:

   INTERV2, INTERVH, K3PH, NUMVAR, and VARINT

9) Update the array, LFDPN, in Routine Y to account for the length of the pattern data changing as a result of the new controller and its associated cycle timing data.

10) Update the following arrays in the PTNRAD subroutine of the off-line program, PATGEN:

CONSEC, ICPOS, HISTLOC, IBYTE.

The reader should consult Section 7. 1 for the procedures to be used in updating these arrays.

11) Update the PATGEN data deck to include phasing data, variable interval data, cycle length, offset, and entry interval data for the new controller.

12) Update the ABCEND and ABCSTRT arrays of Subroutine LLGTHF to reflect the signal phasing of the new controller.

13) If the new controller is not to be controlled by the TANSTP MICRO loop algorithm, LOCAL, enter a zero in the element of the array LOCFLG corresponding to the new controller. This array is contained in Subroutine LINKD.

14) If the new controller has no instrumented approaches and is not to be controlled by the second generation software in the Traffic Responsive Mode, all necessary updates have now been made. The procedures of Sections 8. 2. 1, 8. 2. 2, and 8. 2. 3 should be used to reassemble and relink the software system. (Note: In certain instances it may not be desirable to control a controller under the Traffic Responsive Mode of control. Such instances are an isolated controller not attached to the traffic signal network or a set of controllers which have no instrumentation associated with them. If the above procedures are used to insert these controllers into the data base, the software has been designed so that these controllers will run under the Time-of-Day Mode of control if the operator selects the Traffic Responsive Mode for the section to which the controllers are associated.)

15) If the new controller has instrumented approaches associated with it, the procedures of Section 6. 3. 1 should be used to insert the new links before implementing Step 14), above.

16) If the new controller has critical links associated with it, and it is desired that the new controller be a critical intersection controller, the following arrays require updating before the procedure of Step 14) is implemented:

CICSCT, CICSYS

These arrays are in Routine RTNY and have one bit per controller. The bits associated with the new controller should be set to one if it is desired that this new controller be a critical intersection controller.

17) If the new controller is to be controlled by the second generation software in the Traffic Responsive Mode of control, the following procedures should be used in addition to those of 1) through 16).

18) Establish the group number to which the new controller is to be assigned. This group number assignment is to be reflected in the following arrays of Subroutine LINKD:

CGRPN, GCONT, GCONTSI, GRPNO, NCIGRP.

19) Establish the second generation link numbers associated with the new links which attach the new controller to the traffic signal network. Use the procedures of Section 6. 3. 2 to insert these new second generation links into the data base.

20) If the new controller is to be controlled by the TANSTP MICRO loop algorithm, LOCAL, the second generation link numbers associated with each phase should be inserted into the array LOCDAT of Subroutine GOMTRY. The reader should consult Figure 14 of Section 3. 8 to determine the order in which these links should be entered into this array. The array LOCFLG should be updated to account for any new positioning of the entries in LOCDAT for each controller as a result of adding the new links to LOCDAT.

21) If the new controller is to be controlled by the TANSTP MICRO loop algorithm, LOCAL, the following arrays should be updated to include the new controller before implementing Procedure 14):

ICGSI, INCG, and NICIGP.

6.3.4 Procedures for Changing Group Assignment for a Controller

The procedures for changing the group to which a controller is assigned are as follows:

1) The following arrays require modification:

   GCONT, GCONTSI, GRPNO, ICGSI, INCG, NCIGRP, NICIGP, N2GCL, 2GCLSI.

2) If the controller is on group interface boundary between two groups, the following arrays in Routine INTFC require updating:

   INTN, INTN2, LINKN, NINTF

3) The procedures of Section 8.2.1 should be used to reassemble and relink the software system.

6.3.5 Procedures for Changing Section Assignment for a Controller

The procedures for changing the section to which a controller is assigned are as follows:

1) Employ Procedures 1) and 2) of Section 6.3.4.

2) Update the following arrays in Subroutine PTNRAD of the off-line program PATGEN:

   CONSEC, ICPOS, HISTLOC, IBYTE, LNKST, LNKEND, SECLKN.

3) Update the following arrays in Routine RTNY:

   CONSEC, CICSCT, CTLSEC, LFDPN, LINKSI, NC, SECCONSI.

4) If the controller is on a section interface boundary between two sections, update the following arrays in Routine INTFC:

   INTNS, INTN2S, LINKNS, NGRPIS, NINTFS.

5) Employ the procedures of Sections 8.2.1, 8.2.2, and 8.2.3.

7. OFF-LINE SUPPORT PROGRAMS

There are several off-line programs and card decks associated with the UTCS/TANSTP software system which are required to initialize the system or to dump data collected during system operation. These programs and decks are described in this section. A list of off-line programs and data is given in Table 29. The form of the source is also given in this table.

Table 29. Off-Line Support Programs

Program	Source
PATGEN program	Binary tape
PATGEN data	Card
Predictor coefficient generator (PARAM)	Card
Predictor file allocator	Card
Predictor initialization data	Tape
Surveillance tape dump	Card
Surveillance data	Tape, (generated during real-time execution)
Link and load	Card deck and COC tape
System save	Card

7.1 PATGEN PROGRAM

The PATGEN program is used to generate and store fixed controller timing parameter sets on the system Rapid Access Disc (RAD). These fixed patterns are capable of being selected for implementation through the manual control panel by selecting the Manual or Time-of-Day Mode. The Time-of-Day pattern fixed interval data is also used in the Traffic Responsive Mode as described in Section 2.2.

The PATGEN program is documented in Section 4.5 of Reference 4. As described in Section 6.3, several data arrays need to be changed in Subroutine PTNRAD of this program if the network configuration is changed. The array names, their definitions, and the procedures to be used to determine the values to be entered are given in the following subsections of this document.

7.1.1 PATGEN Program Data Arrays

The array names and their definitions are as follows:

CONSEC - The section number to which each controller is assigned. An entry of 99 indicates that the controller does not exist.

HISTLOC - Byte starting index in pattern data array for each section of controllers. The starting index points to the byte at which the historical volume data begins.

IBYTE - Number of words of pattern data for each section.

ICPOS - Byte starting index in pattern data array for each controller. The starting index points to the byte in pattern data array of the section to which the controller is associated.

LNKST - Position in SECLKN array which contains the first link number in each section.

LNKEND - Position in SECLKN array which contains the last link number in each section.

NSECM1 - Total number of sections of controllers in system minus one..

SECKLN - List of link numbers in system. The entries are grouped by section. That is Section 1 links are listed first, Section 2 second, and so on.

The entries to the CONSEC, LNKST, LNKEND, NSECM1, and SECKLN arrays are straightforward. However, the entries to ICPOS, HISTOC, and IBYTE arrays are not simply determined.

Before giving the procedures for calculating the entries for these latter arrays, the reader should review Figure 2.24.2-2 on Page 2.24-4 of Reference 4. One such pattern data array is reserved for each section of controllers in the system.

From this figure, the entry for IBYTE for Section i is computed as:

$IBYTE_i$ = 8 * Number of controllers in Section i + 7 * number of Critical Intersection controllers in Section i + Number of first generation links in Section i.

The entry for HISTLOC for Section i is computed as

$$HISTLOC_i = 1 + 4 * (IBYTE_i - \text{Number of first generation links in Section } i)$$

The value of the entry in ICPOS and the entry index in ICPOS for the nth controller in the CONSEC array is computed as:

$$ICPOS_{ji} = i + \sum_{k=1}^{n-1} P_k, \qquad j = \sum_{k=1}^{n-1} Q_k$$

where

$ICPOS_{ji}$ is the jth entry for Section i

i = is the value of the entry in the nth position of CONSEC
k is the kth entry in CONSEC

$P_k = 0$ if $CONSEC_k \neq i$

$P_k = 32$ if $CONSEC_k = i$ and controller $_k$ is not a Critical Intersection controller

$P_k = 60$ if $CONSEC_k = i$ and controller $_k$ is a Critical Intersection controller

$Q_k = 1$ if $CONSEC_k = i$

$Q_k = 0$ if $CONSEC_k \neq i$

7.2 SURVEILLANCE DATA DUMP PROGRAM

This program is used to read the 5-minute surveillance data tape and write the contents of the tape to the line printer. The data are printed in the following order for each 5-minute period for which data were collected.

Link number, link fail flag, average link volume in vehicles per hour, average link speed in miles per hour, average link occupancy in percent, free flow detector (Q_2 detector) fail flag, free flow line volume in vehicles per hour, free flow link speed in miles per hour, free flow link occupancy in percent, and one blank parameter.

7.3 PREDICTOR FILE INITIALIZATION PROGRAM

This program is used to initialize the predictor files described in Section 3.3 with the data computed by the Predictor Parameter Generation program. The initialization program is used in conjunction with the data tape generated by the Parameter Generation program. The predictor files

require initialization only if for some reason the contents of the RAD are destroyed. The normal operating procedure is to use the predictor file data from the preceding day. This data is automatically available if a binary save of the RAD was made after the system was shutdown on the previous data and the system is booted in from the binary tape prior to bringing the system on-line. The procedure for making a binary save is contained in Section 4. 3 of Reference 4. The procedure for booting the system from tape is given in Section 4. 1. 2. 2 of Reference 3.

7.4 PREDICTOR PARAMETER GENERATION PROGRAM (PARAM)

The objective of the parameter generation program is to compute the fourier series for volumes and speeds for selected links and create a tape containing this data.

Program PARAM computes Fourier series for twelve links at a time. For each set of links, volume and speed data is extracted from the input tape and stored in an internal array. Because of the uncertainties in the input data, all adjacent volume data points that are identical are eliminated from this array. After this clean-up has been performed, Subroutine FORIER is called to estimate the Fourier coefficients for each link. In order to insure that stray data points will not distort the parameter estimation, all data points that are more than three standard deviations from the Fourier series fit are eliminated from the internal array and the Fourier coefficients are reestimated, via another call to FORIER. The Fourier estimates of volumes and speeds are then computed and stored in an internal array for subsequent output to tape.

The hierarchy of subroutines for the parameter generation system is shown in Figure 80.

The Fourier coefficients are estimated via regression techniques. Subroutine FORIER computes the quantities.

$$\text{Sin } (2\pi it/k)$$

and

$$\text{Cos } (2\pi it/k)$$

for i = 1, 6 and all data points.

These twelve quantities are then passed as the independent variables to Subroutine MLTREG along with volume or speed as the dependent variable.

Subroutine MLTREG is a generalized multiple regression package that simultaneously estimates regression coefficients for m independent variables as a function of n dependent variables as follows:

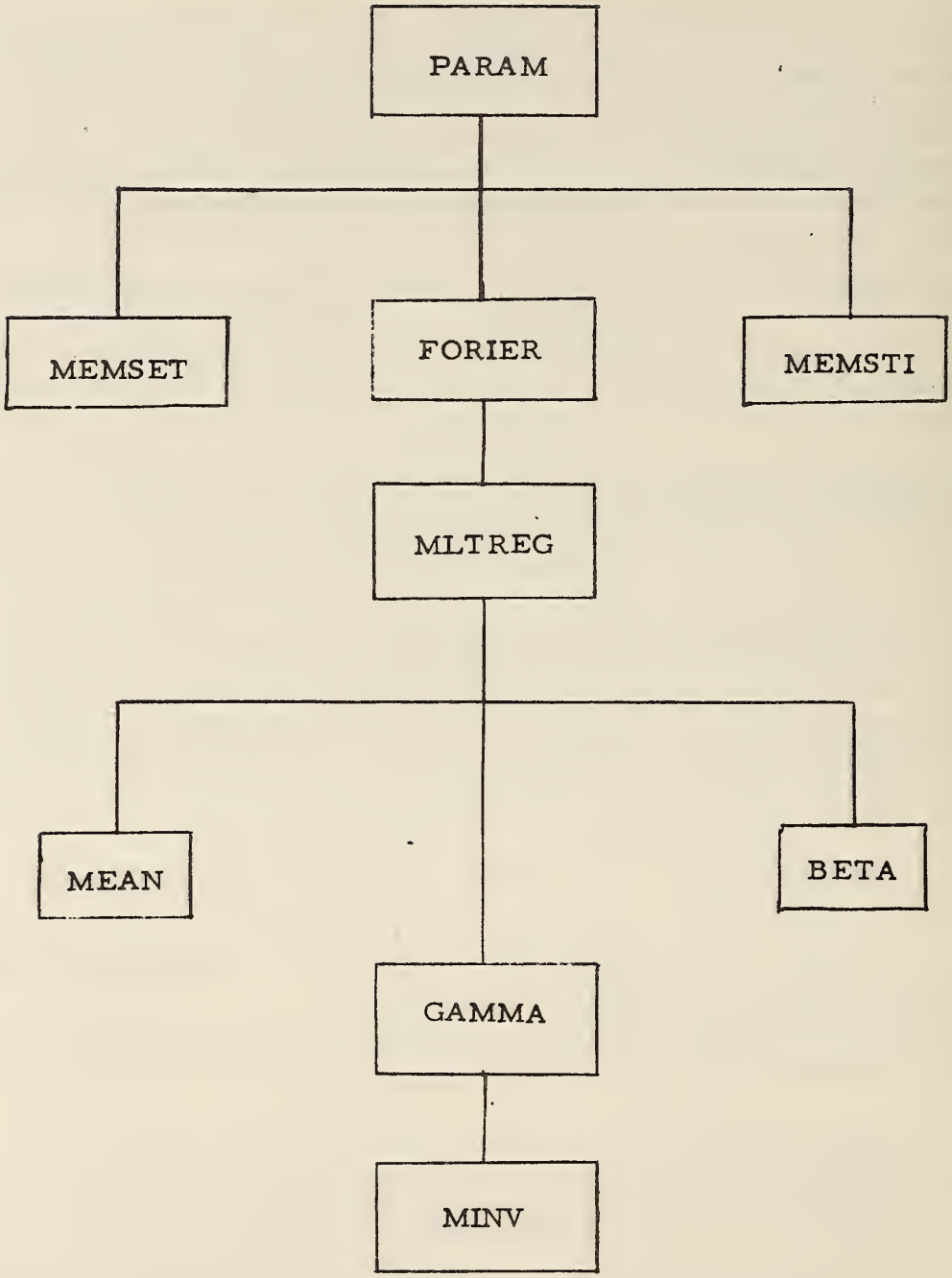


Figure 81. Parameter Generation Program - Hierarchy of Subroutines

Let p_{ij} be the ith data point of dependent variable j. Let q_{ij} be the ith data point of independent variable j. Let $\overline{p}_j$ be the mean of dependent variable j and $\overline{q}_j$ be the mean of independent variable j.

The normalized data are determined from

$$x_{ij} = p_{ij} - \overline{p}_j$$

$$y_{ij} = q_{ij} - \overline{q}_j$$

The regression model is then,

$$X_i = A\ Y_i$$

where

X_i and Y_i are the vectors

$$X_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{in} \end{bmatrix} , \qquad Y_i = \begin{bmatrix} y_{i1} \\ y_{i2} \\ \vdots \\ y_{in} \end{bmatrix}$$

A is an $n \times n$ matrix. The expected value of every element of X_i is zero since X_i represents a vector of normalized residuals.

The matrix A is determined as follows:

Let the vector Z_i be formed by uniting the vectors X_i and Y_i

$$Z_i = \left[\frac{X_i}{Y_i} \right]$$

The elements of Z_i are jointly distributed according to a multivariate normal distribution with zero mean. The covariance matrix for Z_i is Σ, where

$$\Sigma = E \left\{ Z_i\ Z_i^T \right\}$$

and E is the symbol for expection. The Σ matrix may be partitioned into the matrices Σ_{11}, Σ_{12}, Σ_{21}, and Σ_{22} as follows:

$$\Sigma = \left[\begin{array}{c:c} \Sigma_{11} & \Sigma_{12} \\ \hdashline \Sigma_{21} & \Sigma_{22} \end{array}\right]$$

where Σ_{11} is the covariance matrix for X_i and Σ_{22} is the covariance matrix for Y_i.

Since X_i and Y_i are correlated, the generated values of X_i must be distributed according to the multivariate conditional distribution for X_i, given Y_i. The expected value of X_i, given Y_i is

$$E\left\{X_i, Y_i\right\} = \Sigma_{12}\,\Sigma_{22}^{-1}\,Y_i$$

The matrix A is related to these covariance matrices as:

$$A = \Sigma_{12}\,\Sigma_{22}^{-1}$$

Estimates of the covariance matrices may be obtained from the following relationships:

$$\hat{\Sigma}_{11} = S_{11} = \overline{X_i\,X_i^T}$$

$$\hat{\Sigma}_{12} = S_{12} = \overline{X_i\,Y_i^T}$$

$$\hat{\Sigma}_{21} = S_{21} = \overline{Y_i\,X_i^T}$$

$$\hat{\Sigma}_{22} = S_{22} = \overline{Y_i\,Y_i^T}$$

Subroutine BETA estimates the above covariance matrices and Subroutine GAMMA computes the matrix A. Subroutine MINV (see IBM Scientific Subroutine Package) is employed to compute the inverse of Σ_{22}.

The Program PARAM is shown in Figure 82, Subroutine FORIER in Figure 83, Subroutine MLTREG in Figure 84, Subroutine BETA in Figure 85, Subroutine GAMMA in Figure 86, Subroutine MEAN in Figure 87, Subroutine MEMSET in Figure 88, and Subroutine MEMSTI in Figure 89.

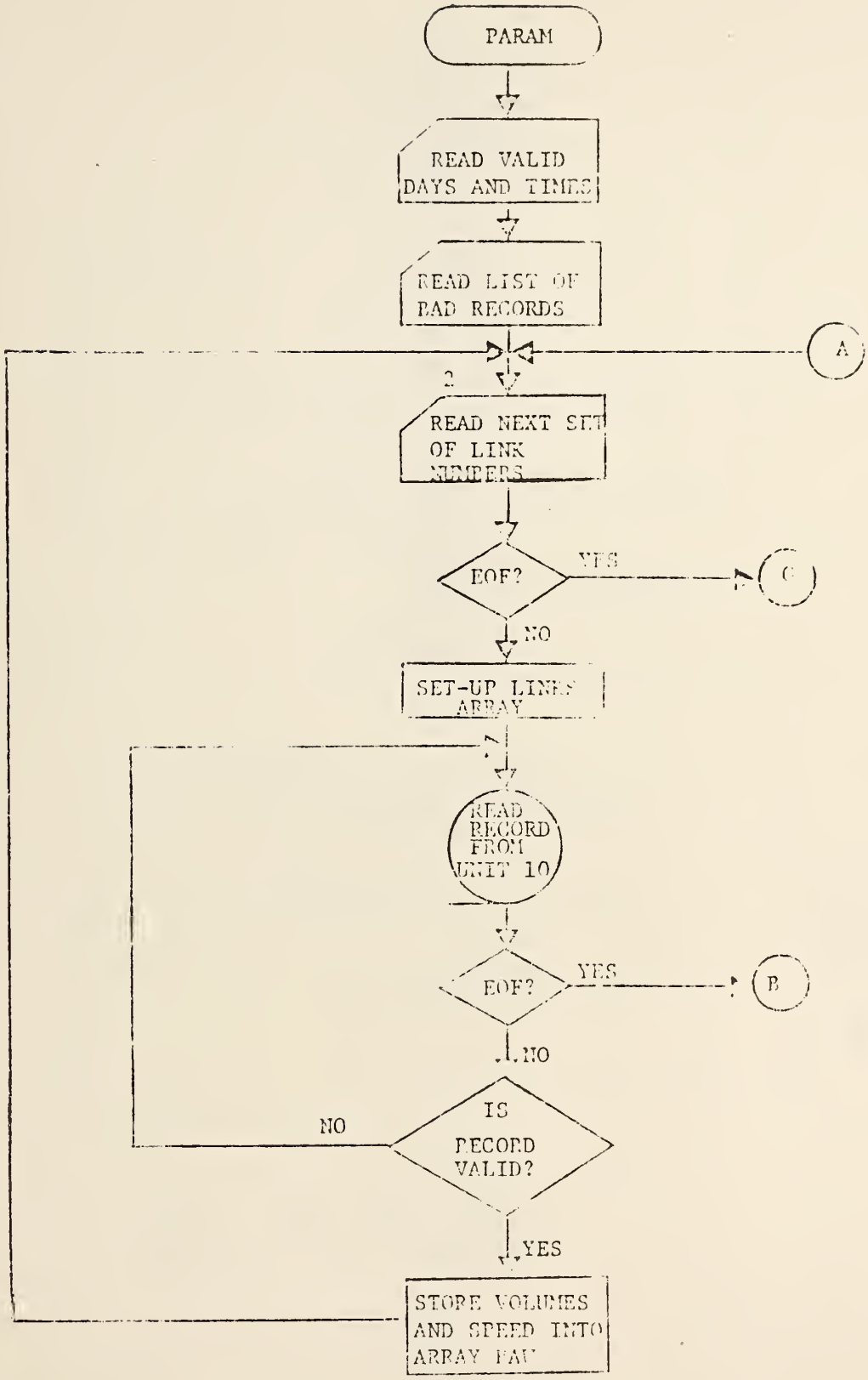


Figure 82. Program PARAM

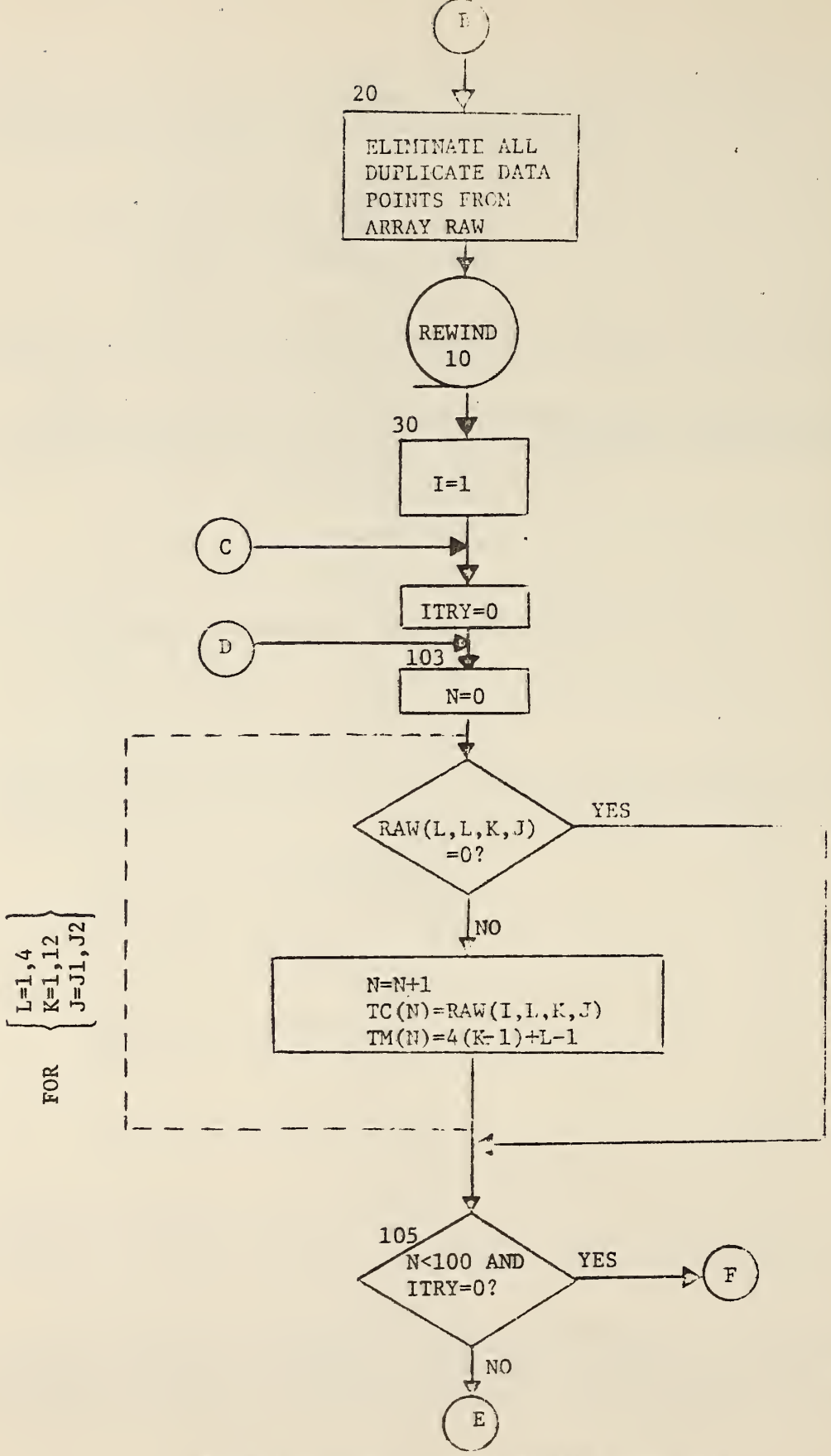


Figure 82. Program PARAM (Continued)

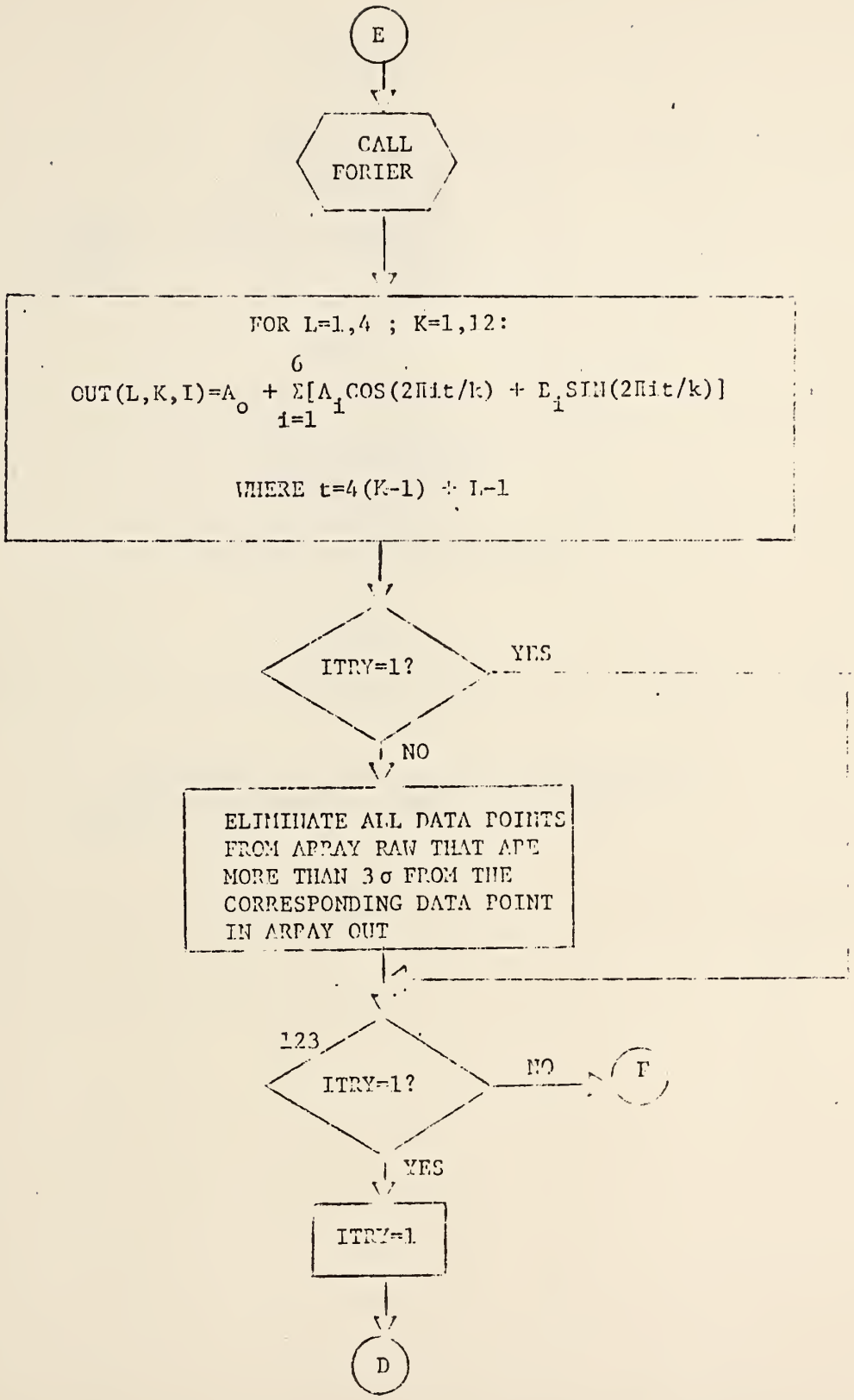


Figure 82. Program PARAM (Continued)

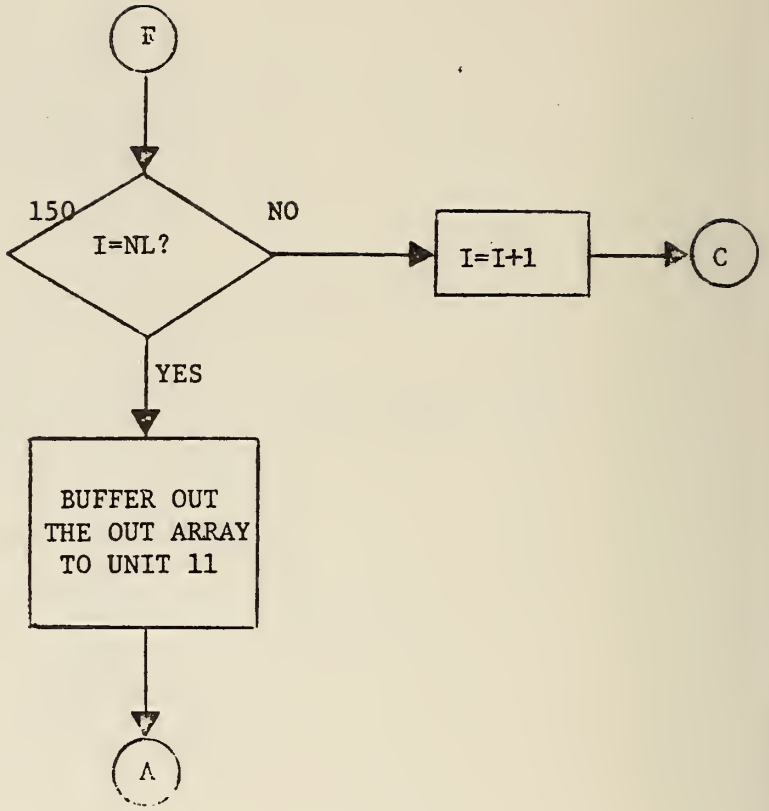


Figure 82. Program PARAM (Continued)

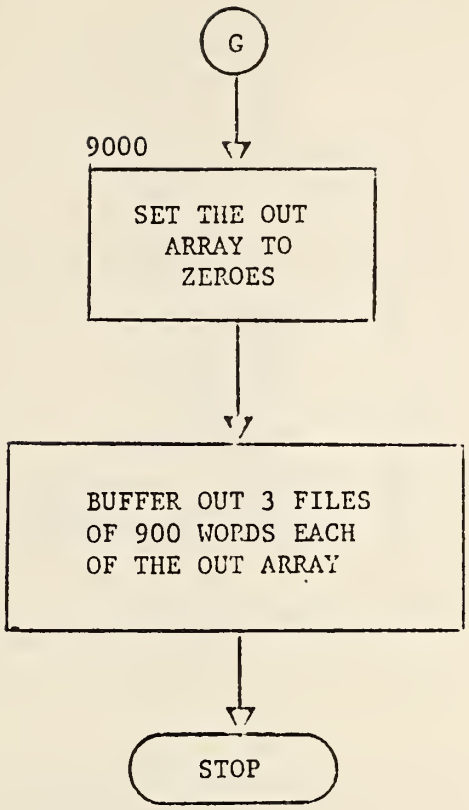


Figure 82. Program PARAM (Concluded)

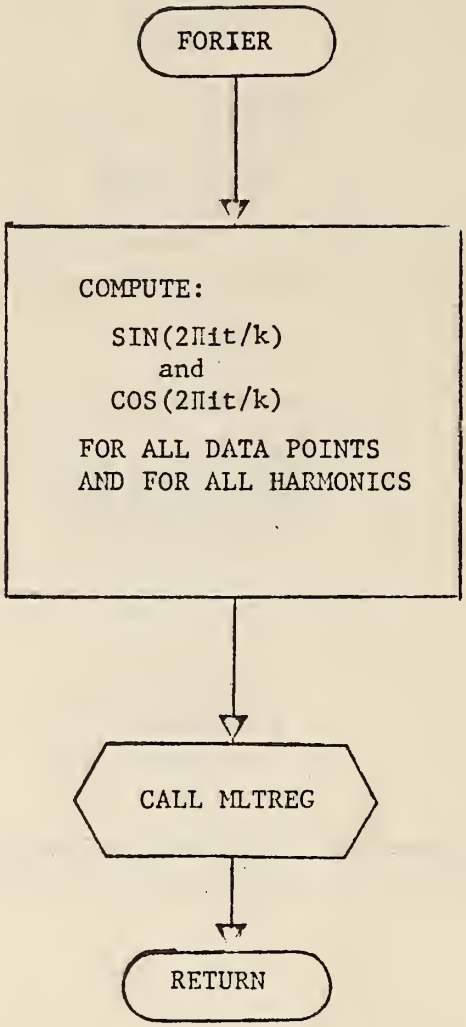


Figure 83. Subroutine FORIER

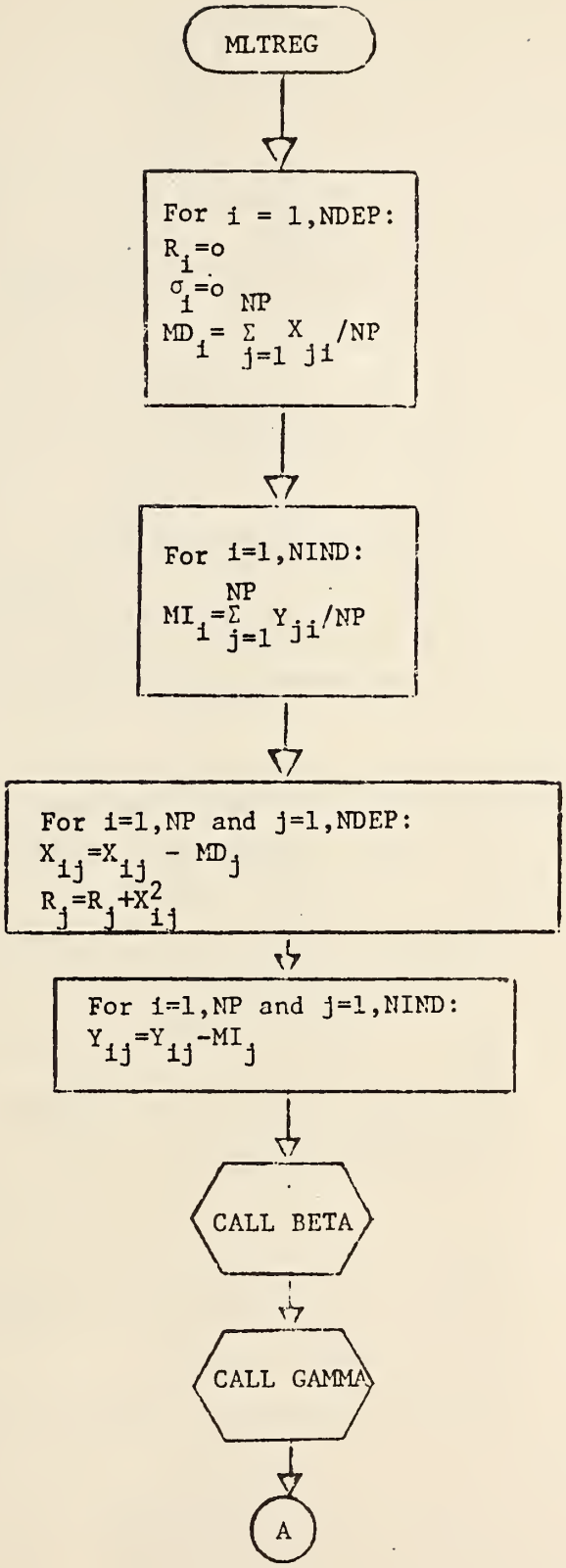


Figure 84. Subroutine MLTREG

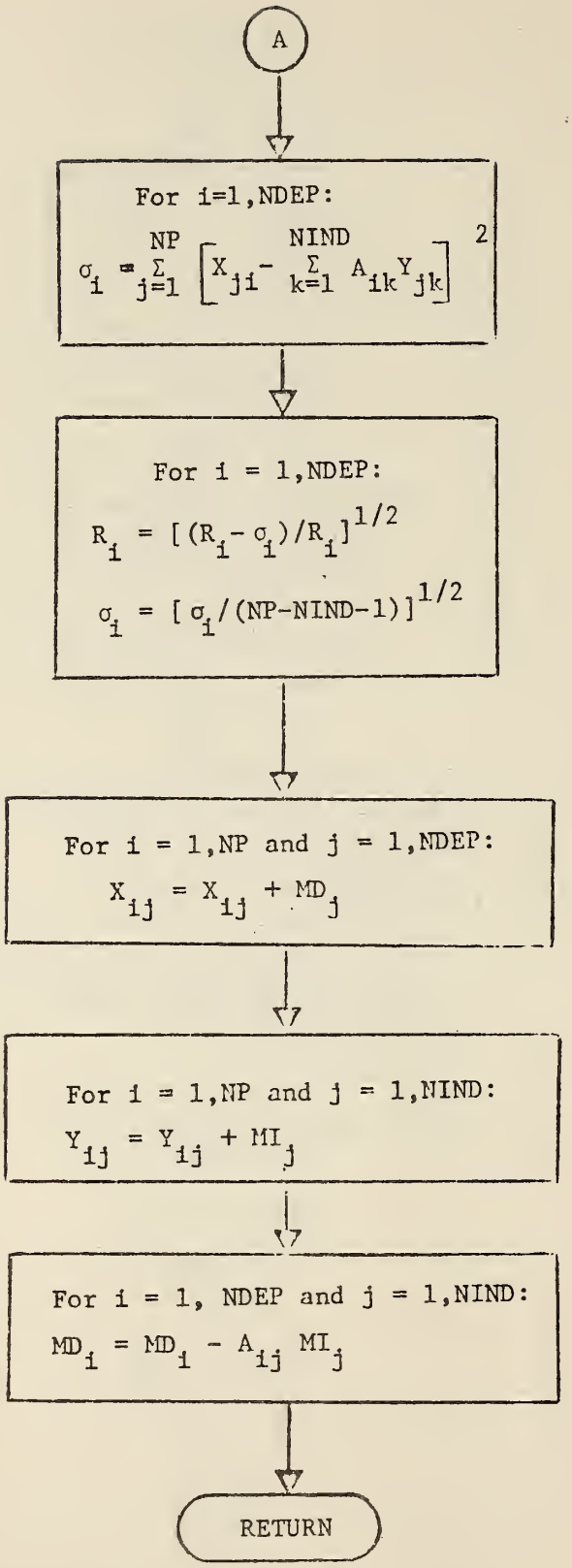


Figure 84. Subroutine MLTREG (Concluded)

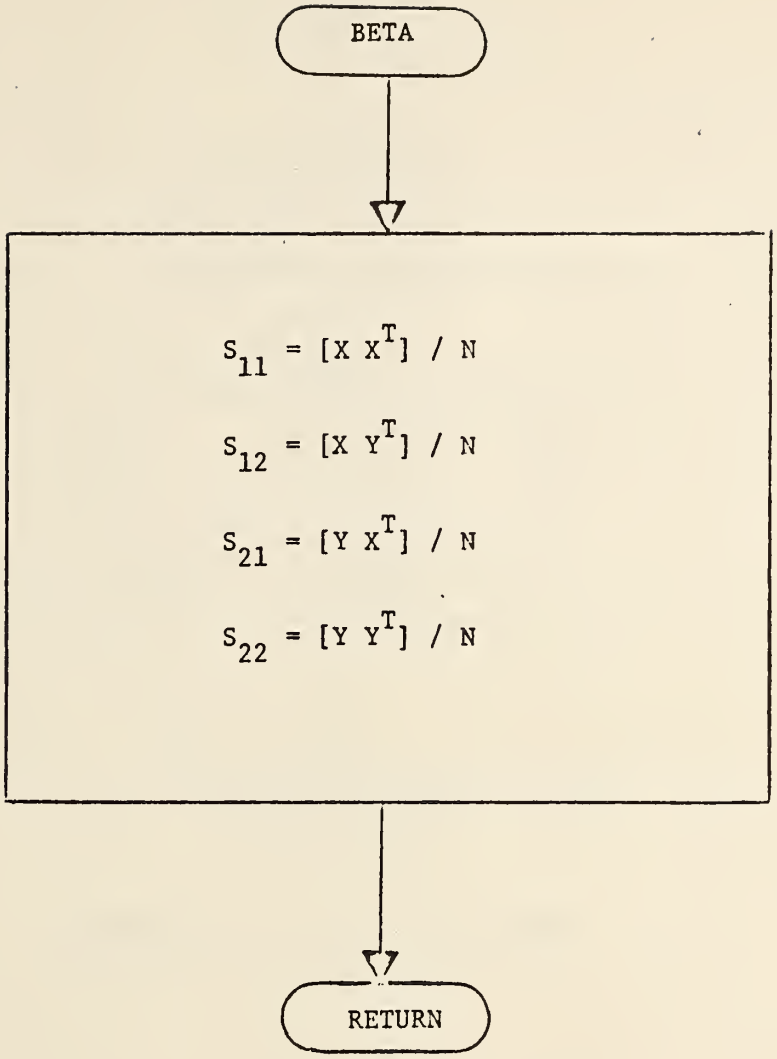


Figure 85. Subroutine BETA

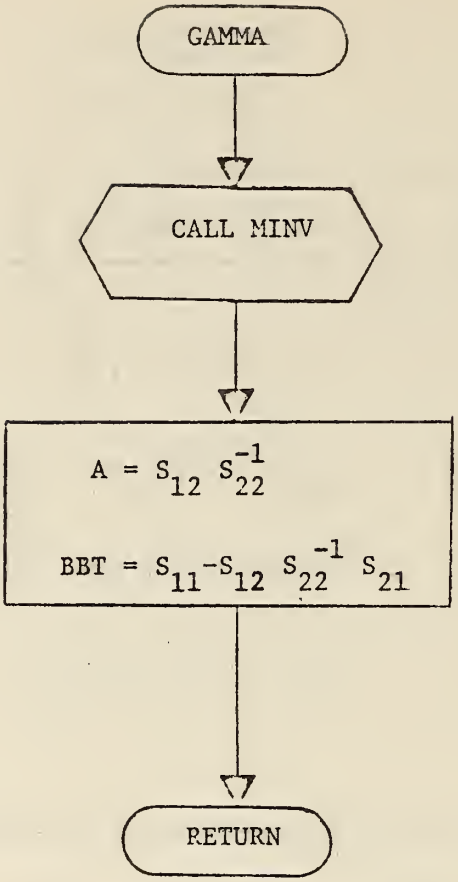


Figure 86. Subroutine GAMMA

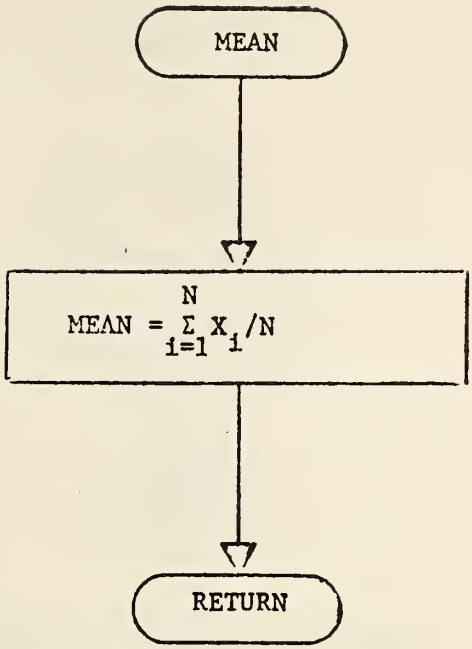


Figure 87. Subroutine MEAN

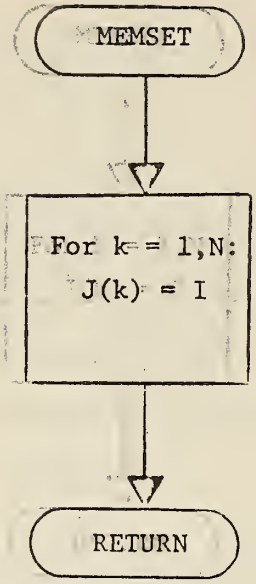


Figure 88. Subroutine MEMSET

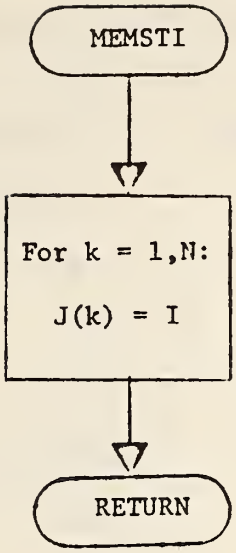


Figure 89. Subroutine MEMSTI

Table 30. Input Data Formats - Program PARAM

Data Name	Dimension	Format	Number of Cards
ITM (I, J)	4, 12	16 (A4, 1X)	1
IDAYS (I, J)	2, 20	20A4	2
NBAD	-	I4	1
DBAD (I, J)	2, 100	A4	NBAD
LBAD (I, J)		A4	
TBAD	100	A4	
LINKS	24	12A4	One for each set of links

Table 31. Definition of Variable Names Used in Main Program PARAM

Mnemonic	Dimension	Definition	Type
AO	-	Constant term in Fourier series	Real
ACON	-	Not used	Real
BCON	-	Not used	Real
CMEAN	20	Mean values of the COEF array (computed by Subroutine FORIER)	Real
COEF	20	Fourier coefficients	Real
DATA (I, J)	2, 255	Input buffer area for tape records J is link index. I=1 → volume I=2 → speeds	Integer
DBAD (I, J)	2, 100	List of bad days on the tape that are not to be used (associated with TBAD)	Integer
G	-	Temporary storage	Real
I	-	Do loop index	Integer
IDAYS (I, J)	2, 20	List of valid days to be used	Integer
II	-	Used by BUFFER OUT	Integer
ISW	-	Temporary storage	Integer
ITM (I, J)	4, 12	List of valid times to be used	Integer
ITRY	-	0 = first Fourier try, 1 = second Fourier try	Integer

Table 31. Definition of Variable Names Used in Main Program PARAM (Continued)

Mnemonic	Dimension	Definition	Type
J	-	Do loop index	Integer
J1	-	Lower index for days to be used	Integer
J2	-	Upper index for days to be used	Integer
J3	-	Not used	Integer
J4	-	Not used	Integer
JJ	-	Do loop index	Integer
K	-	Do loop index	Integer
L	-	Do loop index	Integer
LINKS	24	Current list of link number	Integer
M	-	Do loop index	Integer
MAX	-	Two times the maximum number of links	Integer
MM	-	Maximum number of links that can be processed simultaneously	Integer
MX	-	Two times the number of harmonics	Integer
MXHAR	-	Maximum number of harmonics	Integer
N	-	Do loop index	Integer

Table 31. Definition of Variable Names Used in Main Program PARAM (Continued)

Mnemonic	Dimension	Definition	Type
NBAD	-	Number of bad records on tape	Integer
NHAR	-	Number of harmonics	Integer
NL	-	Two times the number of links that are being processed in the current group	Integer
NMAX	-	Maximum number of data points that can be passed to Subroutine FORIER	Integer
NN	-	Not used	Integer
OUT (I, J, K)	4, 12, 24	Output data area for Unit 11	Real
R	-	Multiple correlation coefficient of Fourier series fit	Real
RAW (I, J, K, L)	24, 4, 12, 12	Area where input data is stored	Real
RWW (I, J, K)	24 48, 12	Equivalenced to RAW	Real
S	-	Temporary storage	Real
SC (I, J)	500, 20	Sines and cosines for the Ith data points	Real
SIG	-	Standard error of the Fourier series fit	Real
T1 (I, J)	20, 20	Temporary storage for FORIER	Real
T2	100	Temporary storage for FORIER	Integer

Table 31. Definition of Variable Names Used in Main Program PARAM (Concluded)

Mnemonic	Dimension	Definition	Type
T3	100	Temporary storage for FORIER	Integer
TBAD	100	List of bad times associated with DBAD	Integer
TC	500	Ith volume or speed that is input to FORIER	Real
TEXT	4	Header data on tape records	Integer
TM	500	Time index of Ith data point that is input to FORIER	Real
TMA	-	Maximum value of time index plus one	Real
X22 (I, J)	20, 20	Temporary storage used by FORIER	Real

Table 32. Definition of Variable Names Used in Subroutine FORIER

Mnemonic	Dimension	Definition	Type
AO	-	Constant term in Fourier series	Real
BBT	-	Used by MLTREG	Real
CMEAN	N	Mean values of the COEF array (computed by Subroutine FORIER)	Real
COEF	N	Fourier coefficients	Real
G	-	Temporary storage	Real
GG	-	Temporary storage	Real
HH	-	Temporary storage	Real
I	-	Do loop index	Integer
J	-	Do loop index	Integer
MXHAR	-	Maximum number of harmonics	Integer
N	-	Number of data points	Integer
NHAR	-	Number of harmonics	Integer
NMAX	-	Maximum number of data points that can be passed to Subroutine FORIER	Integer
PI	-	Constant = 3.141593	Real

Table 32. Definition of Variable Names Used in Subroutine FORIER (Concluded)

Mnemonic	Dimension	Definition	Type
R	-	Multiple correlation coefficient of Fourier series fit	Real
SC (I, J)	NMAX, MXHAR	Sines and cosines for the Ith data point	Real
SIG	-	Standard error of the Fourier series fit	Real
T1	-	Temporary storage for FORIER	Real
T2	100	Temporary storage for FORIER	Integer
T3	100	Temporary storage for FORIER	Integer
TC	N	Data points for dependent variable	Real
TM	N	Data points for independent variable	Real
TMAX	-	Maximum value plus one of independent variable	Real
X11	-	Temporary storage for MLTREG	Real
X12	100	Temporary storage for MLTREG	Real
X21	100	Temporary storage for MLTREG	Real
X22 (I, J)	MXHAR, MXHAR	Temporary storage for MLTREG	Real

Table 33. Definition of Variable Names Used in Subroutine MLTREG

Mnemonic	Dimension	Definition	Type
A (I, J)	MDEP, MIND	Jth regression coefficient for the Ith dependent variable	Real
B	-	Temporary storage	Real
BBT	-	Temporary storage	Real
I	-	Do loop index	Integer
IXY	-	Maximum number of data points (i. e., row dimension of X and Y)	Integer
J	-	Do loop index	Integer
K	-	Do loop index	Integer
MD	100	Constant term in regression for each dependent variable	Real
MDEP	-	Maximum number of dependent variables	Integer
MI	100	Mean value of each independent variable	Real
MIND	-	Maximum number of independent variables	Integer
NDEP	-	Number of dependent variables	Integer
NIND	-	Number of independent variables	Integer
NP	-	Number of data points	Integer
R	NDEP	Multiple correlation coefficient for regression equation for each dependent variable	Real

Table 33. Definition of Variable Names Used in Subroutine MLTREG (Concluded)

Mnemonic	Dimension	Definition	Type
S11 (I, I)	MDEP, MDEP	Covariance matrices computed by Subroutine BETA	Real
S12 (I, J)	MDEP, MIND		Real
S21 (J, I)	MIND, MDEP		Real
S22 (I, I)	MIND, MIND		Real
SIG	NDEP	Standard error of estimate for Ith dependent variable	Real
SUM	-	Temporary storage	Real
T1 (I, I)	MIND, MIND	Temporary storage for Subroutine GAMMA	Real
T2	MIND	Temporary storage for Subroutine GAMMA	Real
T3	MIND	Temporary storage for Subroutine GAMMA	Real
X (I, J)	IXY, NDEP	Ith data point for the Jth dependent variable	Real
Y (I, J)	IXY, NIND	Ith data point for the Jth independent variable	Real

Table 34. Definition of Variable Names Used in Subroutines BETA and GAMMA

Mnemonic	Dimension	Definition	Type
A (I, J)	IM, IMM	Resultant coefficient matrix equal to $S_{11} - S_{12} S_{22}^{-1}$	Real
AA	-	Temporary storage	Real
BB	-	Temporary storage	Real
BBT (I, I)	IM, IM	Not used	Real
CC	-	Temporary storage	Real
D	-	Equal to zero if S_{22} is singular	Real
I	-	Do loop index	Integer
IM	-	Same as M but dimensioned (in the calling program) instead of actual	Integer
IMM	-	Same as MM but dimensioned (in the calling program) instead of actual	Integer
IN	-	Dimensional number of rows of the matrices X and Y in the calling program	Integer
J	-	Do loop index	Integer
K	-	Do loop index	Integer
L	-	Do loop index	Integer
M	-	Actual number of columns of the matrices X, S_{11}, and S_{21}. Actual number of rows of the matrices S_{11} and S_{12}	Integer

Table 34. Definition of Variable Names Used in Subroutines BETA and GAMMA (Concluded)

Mnemonic	Dimension	Definition	Type
N	-	Actual number of rows of the matrices X and Y	Integer
S_{11} (I, I)	IM, IM	Covariance matrix associated with X	Real
S_{12} (J, I)	IM, IMM	Covariance matrix associated with X and Y	Real
S_{21} (J, I)	IMM, IM	Covariance matrix associated with Y and X	Real
S_{22}(I, I)(J, J)	IMM, IMM	Covariance matrix associated with Y	Real
SUM	-	Temporary storage	Real
T1 (I, I)	IMM, IMM	Temporary storage for S_{22} matrix	Real
T2	IMM	Temporary work area	Integer
T3	IMM	Temporary work area	Integer
X (I, J)	IN, IM	Matrix such that the elements of each column have a zero mean	Real
Y (I, J)	IN, IMM	Matrix such that the elements of each column have a zero mean	Real

Table 35. Definition of Variable Names Used in Subroutine MEAN

Mnemonic	Dimension	Definition	Type
I	-	Do loop index	Integer
MEAN	-	Mean value of the vector X	Real
N	-	Number of data points	Integer
SUM	-	Temporary storage	Real
X	N	Vector of data points	Real

Table 36. Definition of Variable Names Used in Subroutine MEMSET

Mnemonic	Dimension	Definition	Type
A	-	Constant value that is stored into each element of the vector B	Real
B	N	Vector that is to be initialized	Real
I	-	Do loop index	Integer
N	-	Length of the vector B	Integer

Table 37. Definition of Variable Names Used in Subroutine MEMSTI

Mnemonic	Dimension	Definition	Type
I	-	Constant value that is stored into each element of the vector J	Integer
J	N	Vector that is to be initialized	Integer
K	-	Do loop index	Integer
N	-	Length of the vector J	Integer

7.5 BINARY SAVE DECK

The Binary Save Deck is used to save the contents of the RAD on magnetic tape. The deck consists of the following Job Control Lanaguage cards. The reader should consult the XDS RBM Reference Manual for definitions of the JCL commands.

```
! JOB

! PAU SYC

! RADEDIT

:SAVE ALL

!FIN
```

This deck should be used at the end of operation for each day to insure that the refined prediction estimates of volume and speed which are generated each day will be available for the next days operation.

7.6 SOFTWARE SYSTEM LINK AND LOAD DECK

The Software System Link and Load Deck is used to link the Relocatable Object Modules (ROM) to form the real time program KESSMANN and load it onto the FP file of the RAD. It also constructs the overlay structure shown in Figure 4. This deck is to be used after any of the routines or subroutines of any of the files listed in Table 6 are updated. The deck consists of the JCL commands listed below. The reader should consult the XDS RBM Reference Manual for a definition of these commands.

```
! JOB

! PAU SYC

! RADEDIT

:DELETE (FIL, FP, KESSMANN)

:SQUEEZE ALL

:ALL (FIL, FP, KESSMANN), (FOR, B), (RSI, 30), (FSI, 2000)

!OLOAD (MAP, ALL), (FORE, 1A00), (TEMP, 1000), (FILE, FP,
                                                KESSMANN);

:(TASK, 9), (UDCB, 10)
```

```
:ROOT (FIL,D4,LOCAL,EOD), (DEV,7TAE1,PACK,EOD);
:(FIL,D4,SUBRT1,EOD);
:(FIL,D4,SUBRT2,EOD);
:(FIL,D4,RTN5F,EOD);
:(FIL,D4,UTCS,EOD), (ENTRY,RTNY)
:SEG (LINK,1,ONTO,0), (FIL,D4,LSQREAL,EOD)
:SEG (LINK,2,ONTO,0), (FIL,D4,RTSND,EOD)
:SEG (LINK,3,ONTO,0), (FIL,D4,PREDICT,EOD)
:SEG (LINK,4,ONTO,0), (FIL,D4,INTFC,EOD)
:ASS (M:DO,LPA02), (VFC)
:ASSIGN (F:108,LPA02), (VFC)
:ASSIGN (F:11,D4,COEFF)
:ASSIGN (F:12,D4,STAT1)
:ASSIGN (F:13,D4,STAT2)
:ASSIGN (F:14,D4,STAT3)
:ASSIGN (F:15,D4,STAT4)
:ASSIGN (F:16,D4,STAT5)
:ASSIGN (F:17,D4,STAT6)
:ASSIGN (F:18,D4,COR)
!FIN
```

8. UTCS/TANSTP SYSTEM OPERATING PROCEDURES AND SOFTWARE SYSTEM UPDATE PROCEDURES

8.1 UTCS/TANSTP SYSTEM OPERATING PROCEDURES

Most operating features and system functions supported by the first generation software are also supported by the TANSTP software. The user should consult Reference 3 for first generation operational procedures. Those features which are different are listed below.

1) The TANSTP system is brought into execution by keying in RUN KESSMANN on the teletype.

2) the MACRO loop is executed every 5 minutes on the 5-minute mark regardless of which system mode is selected. If a nontraffic responsive mode is selected, only the TANSTP surveillance and prediction programs are executed. If a tape (with write ring) is mounted on Unit 7TAE1, the surveillance data is written to this tape every 5 minutes on the 5-minute mark. (It should be noted that the first generation 15-minute tape is still written every 15 minutes on the quarter-hour mark if a tape is properly mounted on Unit 7TAE0.)

3) If on one 5-minute pass, a traffic responsive pattern is generated and implemented, a delay is built into the traffic responsive scheduling logic such that a new pattern will not be generated on the next 5-minute mark. On the subsequent 5-minute mark, a pattern will be generated and implemented only if the reoptimization decision logic of Table 7 indicates that reoptimization should occur. However, the software is designed so that the operator has the capability to override the reoptimization decision logic and force reoptimization to occur. This override can be accomplished while the system is in the Traffic Responsive Mode by merely entering the sequence Traffic Control, System, Traffic Responsive, On-line/Activate, Enter on the system control portion of the manual control panel prior to the subsequent 5-minute mark.

4) The operator is given a 'WAIT' response if he attempts a system or section mode change while the traffic responsive optimization programs are in execution.

5) Since no BPS functions are supported by the TANSTP software, the operator is given an 'ERROR' response if any BPS mode or display is selected.

6) If the 15-minute report is not suppressed, the signal timing patterns in effect at the 15-minute mark will be printed as part of the report. The pattern report consists of the cycle length for each controller and the offsets for each controller phase. If a controller is under second generation control, the subnetwork in which it is operating will be printed as part of the pattern report. A subnetwork number of zero indicates that the controller is not under second generation control. (Note: If Section 4 of the Washington, D. C. network is selected for traffic responsive control, only the controllers on 17th Street and controllers on Constitution Avenue will be optimized by the TANSTP software. The remainder of the controllers in Section 4 will operate in the time-of-day mode.)

7) The predictor executes for the 12-hour period from 7:00 a. m. to 7:00 p. m. During the remaining 12 hours in a day the predictor merely returns the actual volume and speed computed by Routine RTN5 and Subroutine RTNFT. During the 7:00 a. m. to 7:00 p. m. period, the predictor computes corrections to the predicted estimates based upon the variation of the predicted estimates from the actual estimates. These corrections are computed every 5 minutes and stored on the RAD. They are used on the next day to refine the prediction estimates for that day. Because of this feature, the operator should save the system RAD at the ned of operations each day to insure that these corrections are not destroyed.

   CAUTION: If for some reason a particular day had abnormal traffic conditions due to such things as a special event, holiday, accident, etc., the correction terms for that day should not be saved and used the next day. To preclude having to reinitialize the predictor files and starting the refining process over in the event of abnormal operations, it is recommended that two save tapes be used to support the operation of the system. These tapes are to be used on alternate days. That is, Tape 1 will be booted in at the start of Day 1. Tape 2 is used to save the system at the end of Day 1. Tape 2 is booted in at the start of Day 2. Tape 1 is used to save the system at the end of Day 2 and is booted in at the start of operations on Day 3, and so on. These procedures will insure that a backup system is always available in the event the contents of the RAD are inadvertantly destroyed on any given day.

8) A predictor summary report will be printed at midnight and at system shutdown. This report consists of error statistics as described in Section 4.4.2.

9) The MICRO loop control routine, LOCAL, is activated in the same manner as the CIC routine in the first generation software. This routine can be activated regardless of whether the section in which the controller resides is in traffic responsive, time-of-day, or manual. Uner certain conditions, LOCAL may reoptimize the offsets of a controller under MICRO loop control. Therefore, the intersection status display may show a working and actual offset which are different from the entry offset.

8.2 UTCS/TANSTP SOFTWARE SYSTEM UPDATE PROCEDURES

This section provides procedures to be followed if for some reason any part of the software system code or data base requires updating. As stated in Section 2.6, the UTCS system is comprised of the nine files listed in Table 5. For any given modification only those files which contain subroutines which are modified require compilation and/or assembling. Table 6 lists the routines and subroutine contained in each file and the form of the source input for each file. Section 6.3 lists procedures for updating the UTCS/TANSTP data base. The procedures of this section pertain to the sequence which the operator should follow in generating the update software system. The off-line programs and decks referred to below are listed in Table 29 and are described in Section 7.

8.2.1 Procedure to Update File Contained on Cards

1) Boot in binary save of TANSTP from tape.

2) Compile/assemble file to be updated from card reader.

3) Mount COC tape on Unit 7TAE1 and run system link and load deck through card reader. A revised load map will be generated by the system.

4) Mount save tape on 7TAE0 (insert write ring) and run system save deck through card reader. A binary save will be written to tape by the system. The write ring should be removed after the tape has been generated to prevent an inadvertant destruction of the save.

8.2.2 Procedure to Update File Contained on Compressed Source Tape (File UTCS)

1) Boot in binary save of TANSTP from tape.

2) Insert desired updates in update deck, (see Section 4.1 of Reference 4).

3) Mount compressed source tape on Unit 7TAE0.

4) Run update deck through card reader.

5) Mount COC tape on Unit 7TAE1 and run system link and load deck through card reader. A revised load map will be generated by the system.

6) Mount save tape on 7TAE0 (insert write ring) and run system save deck through card reader. After save has been generated, remove write ring from tape.

8.2.3 Procedure to Update Pattern Data Via PATGEN

1) Boot in binary save of TANSTP from tape.

2) Mount PATGEN binary tape on Unit 7TAE0. (Caution. The tape marked "w/o new links in Section 1" should be used until RTNY is updated to accommodate the new detectors which have been placed in Section 1.) This tape is not to be confused with the first generation PATGEN source tape.

3) Any update consisting of a controller type change or phase change requires that Routine RTNY and Subroutine LLGTHF data be updated to be consistent with the PATGEN data.

4) Run PATGEN data deck through card reader.

5) Save new system via using Step 4 of Procedure 8.2.1.

8.2.4 Procedure to Initialize Predictor Data Files

1) Boot in binary save of TANSTP from tape.

2) Mount Predictor Data tape on Unit 7TAE0.

3) Run predictor file allocation deck through card reader.

4) Save new system via Step 4 of Procedure 8.2.1.

REFERENCES

1. Ganslaw, M. J., and Schaake, J. C., "A Volume and Speed Predictor for the UTCS/BPS Second Generation Software," Unnumbered Data Sciences, Inc. Report, June 1973.

2. Irwin, N. A., "SIGOP: Traffic Signal Optimization Program," Prepared by Traffic Research Corporation for Bureau of Public Roads Under Contract Number CPR-11-2862, September 1966.

3. "Urban Traffic Control and Bus Priority Sytem Operator's Manual," Prepared by Sperry Systems Management Division for the Federal Highway Administration Under Contract Number FH-11-7605, September 1972.

4. "Urban Traffic Control and Bus Priority System Software," Volume I, Prepared by Sperry Systems Management Division for the Federal Highway Administration, Under Contract Number FH-11-7605, December 1972.

5. Kessmann, R. W., "Upgraded LOCAL Logic for TANSTP Software," Interoffice Correspondence RWK. 1450.72-56, dated 18 October 1972.

6. Kessmann, R. W., "The Upgraded LOCAL Equations - Revisited," Interoffice Correspondence RWK. 1450.72-64, dated 20 November 1972.

7. Kessmann, R. W., "Coding Specifications, Derivation, and Desk Checks of Revised TANSTP Optimization Algorithm," TRW IOC RWK. 1430.75-17, 21 February 1975.

GPO 910-263

TE 662
.A3
no.FHWA-RD-
76-127 BORROWER

DOT LIBRARY
00177657

FHWA

R&D